

64 Pagine
a colori a sole
L. 8.500

CONTIENE 10 PROGRAMMI

SUPER COMMODORE 64/128

LA RIVISTA JACKSON PER GLI UTENTI COMMODORE 64/128

Anno 6 - Numero 26
L. 8.500 - Frs. 12.75

IN COLLABORAZIONE CON
**COMPUTE's
GAZETTE**

- Home Recording Studio per C64 • Progetto di un programma •
- I problemi della programmazione • Programmazione in LM •
- Guida all'acquisto degli strumenti per programmare • Futurware •
- The Great Arcade Machine, come costruire un gioco Arcade •



TOP GAME PLASMATRON



**GRUPPO EDITORIALE
JACKSON**

AREA CONSUMER

Spedizione in Abb. Postale Gruppo III/70 Taxe Percue (Tassa Riscossa) Milano Ferr. Corr.



GRUPPO EDITORIALE
JACKSON

IN COLLABORAZIONE CON

AMSTRAD

PRESENTA LA GRANDE ENCICLOPEDIA DI INFORMATICA PER RAGAZZI



Con il 1° fascicolo
IL GRANDE POSTER
DEL BASIC

VINCI 30 FAVOLOSI COMPUTER AMSTRAD CPC 6128

- IMPARIAMO A PROGRAMMARE
- COME È FATTO E COME FUNZIONA
- GIOCHI, GIOCHI, GIOCHI
- TECNOLOGIA E APPLICAZIONI
- SI FA... NON SI FA

L'informatica ti appassiona? vorresti approfondirla per studiare, giocare, curiosare? Inizialmente, per potersi orientare, è preferibile una guida giovane ma rigorosa e completa, senza risultare difficile o noiosa, da leggere tutta d'un fiato, corredata da centinaia di illustrazioni a colori.



Aut. Min. Rich.

Questa è LA GRANDE ENCICLOPEDIA DI INFORMATICA PER RAGAZZI. Potrai entrare nei segreti del personal computer: storia, termini, concetti fondamentali, hardware, funzionamento e tanti programmi software in linguaggio Basic. JACKSON, in collaborazione con AMSTRAD, ti svela i segreti dell'informatica in fascicoli settimanali, **nella tua edicola**, a sole lire 2.500, da rilegare in 2 splendidi volumi illustrati.

Ma non è tutto! Assieme al grande poster del Basic a colori, sul primo fascicolo scoprirai come vincere uno splendido computer AMSTRAD CPC 6128.

Sganciati dall'ordinario, entra nello straordinario!

DIRETTORE RESPONSABILE
Paolo Reina

**DIREZIONE, REDAZIONE
AMMINISTRAZIONE,
PUBBLICITA'**
via Rosellini, 12 - 20124 Milano
tel. (02) 6880951/2/3/4/5
TELEX 333436GEJIT I

SEDE LEGALE
via Pietro Mascagni, 14 - 20121 Milano

CONSOZIATA USA
GEJ Publishing Group, Inc.
Los Altos Hills 27910
Roble Blanco 94022 California
Tel. 001/4159492028
Telex. (0023) 49959972
GEJ PUBL SUVL

COORDINAMENTO REDAZIONALE
Angelo Cattaneo

REDAZIONE
Massimiliano Anticoli

HANNO COLLABORATO
Mariano Martin, Cesare Palmieri

IMPAGINAZIONE ELETTRONICA
Piera Loddo

COPERTINA E GRAFICA
Cristina Turra

AREA CONSUMER
PUBLISHER: Filippo Canavese

STAMPA
Grafiche Pirovano S. Giuliano M. (MI)

UFFICIO ABBONAMENTI
via Gasparotto, 15
20092 Cinisello Balsamo (MI)
Tel (02) 61290198-6127212-6122527

DISTRIBUZIONE
SODIP
via Zuretti, 25 - 20125 Milano
Spedizione in abbonamento postale
Gruppo III/70
Prezzo della rivista L. 7500 (cassetta)
Arretrato L. 15000
Prezzo della rivista L. 12000 (disco)
Arretrato L. 24000
Abbonamento
L. 49500 (11 numeri con cassetta)
L. 79000 (11 numeri con disco)
Per l'estero
L. 99000 (11 numeri con cassetta)
L. 158000 (11 numeri con disco)

AUTORIZZAZIONE
con cassetta:
Trib. di Milano n.155 del 5/4/86
con disco:
Trib. di Milano n.47 del 2/2/87

Associato al



Testata aderente al C.S.S.T.
non soggetta a certificazione
obbligatoria in quanto la presenza
pubblicitaria è inferiore al 10%

**GRUPPO EDITORIALE
JACKSON**
AREA CONSUMER

Orizzonti	RUBRICA	4
ABC del BASIC	RUBRICA	5
Tips & Tricks	TRUCCHI	7
Conosci l'Informatica	QUIZ	9
I lettori ci scrivono...	POSTA	10
I libri del mese	RECENSIONI	12
Progettate i vostri programmi	ARTICOLO	13
I problemi di programmazione	ARTICOLO	16
Guida agli strumenti	ARTICOLO	17
Supercommodore Clips	NEWS	21
Divagazioni	RUBRICA	24
Programmazione in l.m.	RUBRICA	25
Plasmatron	SUPERGAME	27
Recensioni	GIOCHI	30
La Superclassifica	TOP TEN	33
Guida all'Input	SERVICE	36
Futurware	ARTICOLO	39
The Great Arcade Machine	UTILITY	43
Home Recording Studio	MUSICA	48
La potenza del BASIC	RUBRICA	52
L'Angolo Geos	RUBRICA	53
Multiview	UTILITY	55
Disc Blitz	GAME	56
Bombardment	GAME	57
128 Animator	UTILITY	58
Le pagine dei Listati	SERVICE	60
Mix II	SERVICE	62

Il Gruppo Editoriale Jackson pubblica anche le seguenti riviste:

ELETTRONICA E AUTOMAZIONE - Automazione Oggi - Meccanica Oggi - EO News settimanale -
Elettronica Oggi - Strumentazione e Misure Oggi
INFORMATICA E PERSONAL COMPUTER - Bit - Informatica Oggi Settimanale - Informatica Oggi - PC Magazine -
PC Floppy - Computer Grafica & Desktop Publishing - NTE Compuscuola - Trasmissione Dati e Telecomunicazioni
TECNOLOGIE E MERCATI - Watt - Media Production - Strumenti Musicali
HOBBY E HOME COMPUTER - Fare Elettronica - Amiga Magazine - Amiga Magazine Games - Commodore Magazine
Games - Videogiochi 64 - Amiga Transactor - Olivetti Prodest User - PC Software - PC Games - 3 1/2 software

Orizzonti

Gli scienziati stanno finalmente incominciando a scoprire i misteri del cervello umano. E' un'impresa difficile (e lunga), ma già parecchi fattori importanti sono stati capiti. Tra questi: la velocità del cervello è dovuta al suo uso del parallelismo e della memoria associativa.

Che figura fa il Commodore 64 quando lo compariamo al nostro cervello? In generale, non troppo brillante. Pensate un attimo alle cose che il nostro cervello fa tutto il giorno e tutta la notte. Pensate a tutte le decisioni che vengono compiute, ma non dimenticate compiti precisi come quello di far battere il cuore o di far funzionare i polmoni. Non tralasciate il continuo controllo dei sensi, e non dimenticate l'inconscio, i sogni vanno ben oltre le facoltà immaginative o persino logiche di qualsiasi programma mai scritto. Infine c'è da considerare la più affascinante forma di pensiero: il pensare di pensare. Dopo questa analisi, il C64 potrebbe sembrare una "bestia" abbastanza patetica, ma spendiamo ancora qualche paragrafo per esaminarlo meglio. Solo comprendendo a fondo i suoi limiti, possiamo apprezzarne le capacità. Primo, bisogna considerare che il C64 è basilamente una macchina single tasking, cioè può fare solo una cosa per volta. Questo significa che, se attaccate il C64 ad una mascella e ad un paio di gambe, non è in grado di camminare e masticare una gomma nello stesso tempo.

Seriale contro parallelo

La maggior parte dei microprocessori "pensa" in seriale (il 6502 del Commodore 64 così come il 68000 dell'Amiga). Ciò vuol dire che il processore esegue un'operazione alla volta. Carica il numero 4, aggiungi il numero memorizzato nella locazione di memoria 49152, memorizza il risultato nella locazione 2048. Questa limitazione viene denominata il "collo di bottiglia di Von Neumann" (gli ingegneri chiamano una struttura a collo di bottiglia quando essa costituisce il fattore limitativo di un'esecuzione). Per esempio, il

I trucchi del cervello

Rhett Anderson

collo è il fattore limitativo della velocità di riversamento quando vuotate una bottiglia di aranciata). E' un brutto aneddoto quello che relega la celebrità di John Von Neumann ad una specie di collo di bottiglia, in realtà John Von Neumann era un genio matematico il cui contributo negli anni '40 e '50 portò all'identificazione dei vantaggi dell'immagazzinamento binario ed al concetto di programma immagazzinato. Prima dell'EDVAC (un computer progettato da Von Neumann), i computer dovevano essere controllati manualmente nella parte elettrica per ogni programma. Più di qualsiasi altro, Von Neumann è responsabile dell'architettura del Commodore 64 e di quasi tutti i computer mai costruiti. E' per insegnamento di Von Neumann che solo recentemente abbiamo incominciato a costruire macchine che si lasciano alle spalle il collo di bottiglia. Il nostro cervello pensa in parallelo. Le informazioni sensoriali vengono raccolte, gli impulsi diretti ai muscoli vengono inviati, il ragionamento logico viene svolto simultaneamente. Un sistema parallelo ne può emulare uno seriale. Voi pensate in seriale quando ragionate in questa maniera: (1) avete in mano una mela e (2) tutte le mele sono frutti, quindi (3) avete in mano un frutto. Il cervello pensa in parallelo perché è costituito da bilioni di neuroni, ed ognuno è in comunicazione con molti altri. I neuroni sono relativamente semplici in confronto al singolo (e molto complesso) elemento di computing usato nel 64. I nuovi computer, come il Transputer e la Connection Machine, si stanno avvicinando al sistema del cervello. Essi funzionano con tanti (non quanto quelli del cervello) semplici (ma non quanto i neuroni) elementi di computing.

Memoria associativa

Perché un computer possa ricordare un fatto deve sapere dove questo fatto è stato immagazzinato. Se non lo sa, dovrà cercarlo nella memoria. Un cervello umano pensa in maniera diversa. Vi può capitare di ricordare l'aspetto della cucina della nonna quando odorate le frittelle. Questo modo di ricordare dipende dalla memoria associativa (memoria per associazione). I ricercatori stanno lavorando sodo per sviluppare sistemi a memoria associativa. I programmi database potrebbero trarre vantaggio dalla memoria associativa per scandagliare enormi database in un batter d'occhio. Quanto possiede il signor Mario Rossi sul suo conto? Anche con 100000 nomi in un database, un sistema di memoria associativa troverebbe il conto del signor Mario in un momento.

Il 64 è poi così ottuso?

Allora, quanto è stupido il C64? In confronto al cervello umano è poco intelligente, ma assolve i compiti per i quali è stato progettato, e li assolve bene. Sfidate il vostro C64 ad una gara di addizione in BASIC. Usando carta e penna, fate l'addizione di cento numeri consecutivi, ad iniziare dal numero 98765. Poi scrivete un programma BASIC che fa la stessa cosa. Il vostro C64 vincerà tranquillamente. Infatti, anche i programmatori principianti potrebbero scrivere il programma, farne il debug, salvarlo su un disco ed azionarlo in un tempo che non sarebbe loro sufficiente neanche per sommare i primi 25 numeri. Scoprirete anche che il C64 è molto più attendibile del vostro cervello. Se riuscite a sommare questi numeri senza fare errori, siete veramente bravi. Il C64 potrebbe far funzionare il programma giorno e notte per interi anni senza commettere uno sbaglio. Ci sono cose che voi potete fare che il vostro C64 non potrebbe mai fare e viceversa. Apprezzerete di più il lavoro congiunto del cervello umano e del computer capendo le limitazioni e le possibilità di entrambi.

ABC del BASIC

Io progetto prodotti per una fabbrica di materiale elettrico. Essendo la ditta di livello internazionale, le parti dei nostri strumenti sono dimensionate secondo il sistema metrico. Questo sistema di misurazione, come probabilmente tutti sapete, è usato quasi esclusivamente, fuori dagli Stati Uniti e dalla Gran Bretagna, che utilizzano ancora il sistema inglese. Pur essendo cresciuto con il sistema inglese, io preferisco quello metrico.

Il programma che incominceremo a scrivere questo mese, converte le più comuni unità di lunghezza da un sistema all'altro. Ovviamente, potremmo convertire l'area, il volume, il peso, la massa, la temperatura, e molte altre unità di misura. Ma per non allungare troppo il programma, ci concentreremo sulle conversioni di lunghezza. Nel sistema inglese, le più familiari unità di lunghezza sono il pollice (in.), il piede (ft.), la iarda (yd.), e il miglio (mi.). Nel sistema metrico, le più comuni sono i millimetri (mm), i centimetri (cm), i metri (m) e i chilometri (km).

Come iniziare

Iniziamo col definire il colore del testo. Mi piace il bianco; è molto leggibile e sta bene con quasi tutti i colori di sfondo. Ecco un modo per stampare caratteri bianchi:

```
10 PRINTCHR$(5):REM
   CARATTERI BIANCHI
```

CHR\$() quando viene usato in congiunzione con PRINT, può stampare, cambiare l'aspetto, o anche togliere i caratteri sullo schermo, ed anche muovere il cursore. Perché questa funzione possa essere valida, tra parentesi dovete mettere un numero tra 0 e 255. Ovviamente, CHR\$() ha molti altri usi oltre a quello del PRINT.

Sulla maggior parte dei computer, il codice ASCII da 65 a 90 rappresenta le lettere dell'alfabeto. Ed è così, che quando digi-

**Questo mese
prendiamo in esame
alcuni semplici
ma importanti argomenti
del BASIC.**

**Tratteremo CHR\$(),
minuscole e maiuscole,
e scritte d'array calcolate.**

**Tutto questo
dimostrato con un solo,
elementare, programma.**

Larry Cotton

tate PRINT CHR\$(65) e premete RETURN, compare una A sullo schermo. Altri codici ASCII stampano numeri o simboli, muovono il cursore e cambiano il colore dei caratteri.

Per esempio, quando il numero tra parentesi è 5 (come nella nostra linea di programma più sopra), i caratteri vengono stampati in bianco. Se fosse stato 158, i caratteri sarebbero stati gialli. (Consultate la guida utenti per la lista completa dei codici ASCII e CHR\$).

Ora cambiamo il colore di sfondo dello schermo in nero, così che contrasti con il colore del testo. Ciò non può essere fatto usando CHR\$(); invece, necessita di una POKE, un comando che dobbiamo ancora vedere:

```
20 POKE53281,0:REM
   SCHERMO NERO
```

Naturalmente, se non vi piacciono questi colori, potete cambiarli a vostra discrezione.

Le costanti

Ora, dovremmo pensare a come il programma dovrà funzionare. Vogliamo che l'utente inserisca dei valori che verranno

moltiplicati da costanti (cosiddette fattori di conversione) per ottenere altri valori. Se convertiamo ognuna delle 4 unità inglesi in 4 unità metriche, avremo bisogno di 16 costanti.

Mettiamole in un array di 16 elementi che dimensioniamo con DIM (ricordate che DIM definisce o dichiara il numero di elementi in un array).

```
30 X=16:DIMC(X)
```

Poi, stabiliamo un ciclo FOR-NEXT per leggere le costanti nell'array:

```
40 FORN=1TOX:READC(N):
   NEXT
```

Abbiamo bisogno di alcuni dati per leggere. Ho calcolato personalmente i fattori di conversione per voi; sono tutti basati sul pollice che equivale esattamente a 25,4 millimetri. Iniziamo le nostre istruzioni di DATA alla linea 1000 e rinumeriamo il programma più tardi:

```
1000 DATA .0000254,.0254,
      2.54,25.4
1010 DATA .0003048,.03048,
      30.48,304.8
1020 DATA .0009144,.9144,
      91.44,914.4
1030 DATA 1.609344,
      160.9344,16093.44,
      160934.4
```

Il programma, a questo punto, è pronto per funzionare. Se infatti proviamo a dare il RUN, vedremo che, dopo una breve pausa, apparirà il messaggio READY.

Per esaminare un elemento del nostro array battete PRINT C(N) sostituendo a N un numero compreso tra 1 e 16. Provate vari valori per vedere cosa succede.

Alcune costanti potranno sembrarvi un pò strane, ma questo è un altro argomento, di cui parleremo in dettaglio il prossimo mese.

Il menu

Ora che abbiamo depositato al sicuro i nostri gruppi di data nelle loro 16 caselle, creiamo un menu di opzioni. Iniziamo col cancellare lo schermo, avremo così un'altra possibilità di usare CHR\$():

```
50 PRINTCHR$(147):REM
   CANCELLA SCHERMO
```

Il codice ASCII 147 rappresenta un clear di schermo nei computer Commodore. Ora il menu:

```
60 PRINT"(1) IN TO KM
70 PRINT"(2) IN TO M
80 PRINT"(3) IN TO CM
90 PRINT"(4) IN TO MM
100 PRINT"(5) FT TO KM
110 PRINT"(6) FT TO M
120 PRINT"(7) FT TO CM
130 PRINT"(8) FT TO MM
140 PRINT"(9) YD TO KM
150 PRINT"(10) YD TO M
160 PRINT"(11) YD TO CM
170 PRINT"(12) YD TO MM
180 PRINT"(13) MI TO KM
190 PRINT"(14) MI TO M
200 PRINT"(15) MI TO CM
210 PRINT"(16) MI TO MM
220 INPUT"{DOWN}
    QUALE OPZIONI";N
```

Fate girare nuovamente il programma, apparirà il menu e vi verrà chiesto di inserire un numero che indica la conversione che volete fare.

Cosa succede se l'utente batte un numero che non ha senso? Parecchi IF-THEN avranno il compito di eliminare i numeri inaccettabili:

```
230 IFN<1ORN>16THEN50
```

Se viene battuto un numero fuori dalla scala prevista, queste linee mandano il controllo indietro alla linea 50, che cancella lo schermo e ristampa il menu. Ora chiediamo all'utente di battere il numero delle unità che desidera convertire:

```
240 INPUT"{DOWN}
    QUANTITA' ";Q
```

Elegante ed amichevole

Bene, ciò compie il lavoro, ma manca di due cose: l'user-friendly e l'intelligenza. Come possiamo rendere il nostro programma più elegante? Il computer sa quali conversioni desidera l'utente dal numero che questi inserisce. Per esempio se è 4, l'utente vuole convertire un numero di pollici in un numero equivalente di millimetri. Cambiamo allora la linea 240:

```
240 INPUT"{DOWN}QUANTITA'
    IN POLLICI";Q
```

Funziona magnificamente! Ono? Mettiamo che l'utente chieda una conversione da piedi a chilometri. Potremmo aggiungere alcuni IF-THEN per controllare tutte le possibilità, ma c'è un modo più facile. Facciamo di IN, FT, YD e MI delle costanti di stringa e delle abbreviazioni del sistema metrico altre costanti. Poiché sono associate con le costanti di conversione che abbiamo già stabilito come DATA...ma ci stiamo allontanando troppo nel tentare di rendere il programma più compatto? Non credo. Guardate cosa possiamo fare con pochi cambiamenti. Torniamo indietro e variamo la 30 per dimensionare altre due array di 16 elementi:

```
30 X=16:DIMC(X),IS(X),M$(X)
```

Ora aggiungiamo le unità associate alle costanti di conversione alle linee DATA. In altre parole, la prima costante, .0000254, converte i pollici in chilometri. Non è necessario battere nuovamente le linee; basta inserire alcuni spazi ed aggiungere i data extra.

```
1000 DATA .0000254,IN,KM,
      .0254,IN,M,2.54,IN,CM,
      25.4,IN,MM
1010 DATA .0003048,FT,KM,
      .03048,FT,M,30.48,FT,
      CM,304.8,FT,MM
1020 DATA .0009144,YD,KM,
      .9144,YD,M,91.44,YD,
      CM,914.4,YD,MM
1030 DATA 1.609344,MI,KM,
      169.9344,MI,M,16093,
      44,MI,CM,160934,4,
      MI,MM
```

Notate attentamente l'ordine nel quale i 16 gruppi di data sono disposti: una costante numerica seguita da due costanti di stringa. La linea 40 va variata per leggere i DATA esattamente in quell'ordine:

```
40 FORN=1TOX:READC(N),
   IS(N),M$(N):NEXT
```

Ora possiamo cambiare la linea 240 per usare la nostra amichevole costante di stringa:

```
240 PRINT"{DOWN}QUANTITA'
    DI"IS(N);:INPUTQ
```

Nato per funzionare

Una buona idea quando si scrive un programma BASIC è quella di salvarlo quando può funzionare, se possibile. Così, quando ritornate più tardi, potete più facilmente seguire ciò che avete fatto sino a quel punto. Anche se il nostro programma non è proprio finito, può ugualmente funzionare. Se lo azionate vedrete che le linee DIM e FOR-NEXT funzionano propriamente e che il menu stampa in caratteri bianchi su sfondo nero. Dopo aver dato il RUN, potete anche provare tutte le opzioni del menu battendo PRINT C(N), IS(N), M\$(N), dove N è tra 1 e 16. I valori non devono pareggiare quelli nelle corrispondenti posizioni nelle linee DATA. Ricordate che i DATA sono sistemati in 16 gruppi da tre elementi: il fattore di conversione e le sue due abbreviazioni di unità.

Quasi fatto

Siamo arrivati ad un buon punto: il programma finale così come l'ho scritto io, converte le dimensioni della lunghezza in entrambi i modi (da inglese a metrico e viceversa) ma usa ancora solo 16 costanti. Provate a finirlo voi...un suggerimento: il menu deve raddoppiare nelle dimensioni (tenetelo su uno schermo solo), e le linee 250 e 240 possono ospitare nuove opzioni. Dato che ci siamo, ecco un altro dato per i galattici: il metro è esattamente uguale a 1650,763.73 lunghezze d'onda nel vuoto della radiazione corrispondente alla transizione tra il livello di $2p^{10}$ e $5d^5$ dell'atomo kripton 86!!

Tips & Tricks

A cura di M. Anticoli

48

CALENDARIO DEL MESE

Ogni tanto capita di dover consultare un calendario per vedere la data del proprio anniversario o del compleanno; perchè non usiamo il nostro beneamato C64-128, senza dover ricorrere ai fastidiosi calendari cartacei?

Dopo aver dato il RUN, bisogna introdurre l'anno ed il mese, una volta mostrato il calendario premendo RETURN viene visualizzato il calendario del mese precedente.

```
100 REM CALENDARIO MENSILE
110 POKE59468,14:R$=
CHR$(13):DIMDS(11),M$(11),D(11),L$(23)
120 FORI=0TO11:READDS(I),M$(I):NEXTI
150 INPUT"(CLR){CUR.GIU}ANNO?:IFY=0 THENEND
160 IFY/4=INT(Y/4) THEN DS(1)=29+(Y/100=INT(Y/100)ANDY/400>INT(Y/400))
170 X=Y-1:X=X+INT(X/4)+INT(3*INT(X/100)/4):D(0)=7*INT(X/7)-X
180 FORI=0TO10:X=DS(I)-D(I):D(I+1)=7*INT(X/7)-X:NEXTI
1000 INPUT"(CUR.GIU)MESE?(0=ANNO NUOVO)";M:M=M-1:IFM<0 THENRUN
1010 IFM>11ORM>INT(M)GOTO1000
1020 M$=M$(M):X$=" "+M$+" "+STR$(Y)+" ":X=LEN(X$)
1030 PRINT"(CLR)"SPC(20-X/2);:FORI=1TOX:PRINT{RVS OFF}{CBM@}{CUR.GIU}{CUR.SIN}{RVSON}"MID$(X$,I,1)"{CUR.SU}";:NEXTI:PRINTPRINTSPC(10)"{2 CUR.GIU}DO LU MA ME GI VE SA"
1040
1050 D=D(M):FORL=0TO5:L$="":PRINTSPC(9);:FORI=0TO6:D=D+1
1060 IFD<=0ORD>DS(M)
```

```
1070 THENX$="{3SPC}":GOTO1100
X$=STR$(D):IFD<10THENX$=" "+X$
1100 PRINTX$;:NEXTI:PRINT:NEXTL:GOTO1000
50000 DATA31,"GENNAIO",28,"FEBBRAIO",31,"MARZO",30,"APRILE",31,"MAGGIO"
50005 DATA30,"GIUGNO",31,"LUGLIO",31,"AGOSTO",30,"SETTEMBRE"
50010 DATA31,"OTTOBRE",30,"NOVEMBRE",31,"DICEMBRE"
```



49

DIMOSTRATIVO IN 80 COLONNE

Il C128 è un computer che può funzionare sia in 40 che in 80 colonne mentre il piccolo C64 funziona solo in 40 colonne. In commercio sono stati presentati molti programmi per farlo funzionare in 80 colonne, ad un prezzo abbastanza alto. Il trick che vi presentiamo mostra il nostro C64 ad 80 colonne in un tempo spaventosamente veloce!! Per scrivere qualsiasi cosa in 80 colonne bisogna modificare la variabile A\$ alla linea 126.

```
1 DATA162,0,142,0,4,173,4,192,201,7,208,8,173,3,192,201,255,208,1
2 DATA96,238,3,192,208,3,238,4,192,208,228
5 BA=2*4096:POKE53272,PEEK(53272)OR8
6 L=12*4096:FORI=LTOL+29:READX:POKEI,X:NEXT:GOTO10
7 CH=INT(X/8):RO=INT(Y/8):LN=YAND7:BY=BA+RO*320+8*CH+LN:BI=7-(XAND7)
8 POKEBY,PEEK(BY)OR(2ÉBI):RETURN
9 CH=INT(X/8):BY=11714+8*CH:BI=7(XAND7):GOTO8
```

```
10 POKE53265,PEEK(53265)OR32
20 SYSL
30 POKEI+1,0:POKEI+3,0:POKEI+4,32:POKEI+9,63:POKEI+16,63:SYSL:REMLCLR BATOBA+7999
125 POKE56334,PEEK(56334)AND254:POKEI,PEEK(1)AND251
126 A$="{SH Q}UESTA E' UNA PROVA DI SUPERCOMMODORE!!":FORZ=1TOLN(A$)STEP2
127 T=ASC(MID$(A$,Z)):GO SUB300:U=T:T=ASC(MID$(A$,Z+1)):GOSUB300:GOSUB200:NEXTZ
129 POKEI,PEEK(1)OR4:POKE56334,PEEK(56334)OR1130 GETA$:IFA$=" "THEN130
135 PRINT"(CLR)";
140 POKE53265,PEEK(53265)AND(255-32)
150 POKE53272,PEEK(53272)AND(255-8)
160 END
200 FORI=0TO7
210 A=PEEK(I+U*8+55296):B=PEEK(I+T*8+55296):V=BA+I
220 C=(AAND1)+(AAND4)/2+(AAND16)/4+(AAND64)/8
222 D=(AAND2)/2+(AAND8)/4+(AAND32)/8+(AAND128)/16:A=D
230 C=(BAND1)+(BAND4)/2+(BAND16)/4+(BAND64)/8
232 D=(BAND2)/2+(BAND8)/4+(BAND32)/8+(BAND128)/16:B=D
236 A=A*16+B
240 POKEV,PEEK(V)OR(255-A)
250 NEXT
255 POKEBA/8,1
260 BA=BA+8:RETURN
300 IFT<33THENT=32:RETURN
305 IFT<64THENTRETURN
310 IFT<96THENT=(T)AND63:RETURN
```



```

320  IFT<128THENT=T-32
      :RETURN
330  IFT<161THENT=32:RETURN
340  IFT<192THENT=T-64
      :RETURN
350  IFT<224THENT=T-96
      :GOTO300
360  IFT<254THENT=T-64
      :GOTO300
370  T=126:GOTO300

```



50

UN MONDO AL CONTRARIO

Fare degli scherzi agli amici è molto divertente, con questo mini programma riuscirete a far prendere "un colpo al cuore" ad un vostro amico che magari ha appena toccato il vostro computer!!

```

10  DIM Q(8)
31  POKE 56334,PEEK
    (56334)AND254:POKE1,
    PEEK(1)AND251
35  FOR I = 0 TO 255
36  FOR J = 0 TO 7
37  POKE12288+I*8+J
    ,PEEK(53248+I*8+J)
38  NEXTJ:NEXTI
39  POKE1,PEEK(1)OR4
    :POKE56334,PEEK(56334)OR1
40  POKE 53272,(PEEK
    (53272)AND240)+12
45  FOR X = 12288 TO
    12504+2040 STEP 8
50  FOR W = 0 TO 7
60  Q(W)=PEEK(X+W)
70  NEXTW:C=-1
80  FOR T = 7 TO 0 STEP -
    1:C=C+1
90  POKEX+T,Q(C)
100  NEXT T
110  NEXT X

```



51

NUMERI RANDOM

Il computer è sì un preciso calcolatore, ma ogni tanto anche lui può dare i "numeri". Il programma estrae tre numeri casuali per dieci estrazioni alla volta.

```

0  PRINT "{CLR}":PRINT
1  PRINT "{CLR}"TAB
    (13)"NUMERI CASUALI"
2  FORP=1TO1500:NEXT
5  FORT=1TO10:PRINT"
    {CUR.GIU}"

```

```

10  FORX=1TO3
20  PRINTINT(RND
    (X<0)*10);TAB(3);
30  NEXTX
32  PRINTTAB(12);"NUMERO"
    ;T;
35  NEXTT
36  PRINT:PRINT"{CUR.SU}"
37  PRINT"{6 SPC}{RVS
    ON}PREMI SPACE PER
    CONTINUARE!"
40  GETA$:IFA$=""THEN40
45  IFA$="" THENPRINT"
    {CLR}":RUN5
50  GOTO40

```



52

LISTA VARIABILI

Molto spesso è utile avere a portata di mano la lista delle variabili del programma in cui state effettuando il debug, se non siete dei perfetti programmatori che documentate tutto su fogli di carta questo è il programma che fa proprio al vostro caso. Caricate il programma LISTA VARIABILI, una volta in memoria date il RUN, caricate e fate partire il programma desiderato, alla fine del programma digitate l'istruzione SYS apparsa sullo schermo al RUN del primo programma ed ecco fatto, tutte le variabili visualizzate appaiono sullo schermo!!

```

120  ME=PEEK(55)+
    256*PEEK(56)
130  VS=ME-256:PA=0
140  POKE56,PEEK(56)-1
150  FORI=VSTOVVS+240:
    READA:POKEI,A:PA=PA+A
    :NEXT
160  IFPA<>33670THEN
    PRINT"ERRORI NEI
    DATI":END
170  PRINT"SYS"VS" DOPO
    AVER DATO IL RUN AL
    PRG DI RICERCA":NEW
180  DATA165,45,197,47,
    240,93,133,253,165,46,
    133,254,160,0
200  DATA169,0,141,61,3,
    177,253,41,128,208,60,
    177,253,41
220  DATA127,32,210,255,
    200,173,61,3,201,0,208,
    6,177,253
240  DATA41,128,208,46,177
    ,253,41,127,32, 210,
    255,173,61,3
260  DATA32,210,255,169,
    32,32,210,255,152,24,
    105,6,144,5

```

```

280  DATA164,254,200,132,
    254,168,101,253,197,47,
    240,17,208,186
300  DATA96,169,37,141,
    61,3,208,189,169,36,141,
    61,3,208
320  DATA203,165,49,197,
    47,240,114,165,47,133,
    253,165,48,133
340  DATA254,160,0,169
    ,0,141,61,3,177,253
    ,240,216,41,128
360  DATA208,77,177,
    253,41,127,32,210,255
    ,200,173,61,3,201
380  DATA0,208,6,177,
    253,41,128,208,63,177,
    253,41,127,32
400  DATA210,255,173,61,
    3,32,210,255,169,40,
    32,210,255,169
420  DATA41,32,210,255,
    169,32,32,210,255,
    200,177,253,24,101
440  DATA253,197,49,240,
    39,177,253,24,101,253,
    170,200,177,253
460  DATA101,254,133,254,
    134,253,208,165,96,169,
    37,141,61,3
480  DATA208,172,169,36,
    141,61,3,208,186,165,
    48,197,50,208
500  DATA136,96,200,234,
    177,253,101,254,197,
    50,240,224,16,222,136,
    208,202

```



53

ALFABETO RUSSO

Chissà quante volte avrete visto in TV o in illustrazioni varie le strane scritte dell'alfabeto cirillico usato dai Paesi dell'Est Europeo. Ora col programmino che segue potrete eseguire la trasformazione dei caratteri normali in quelli dell'alfabeto cirillico.

```

5  PRINTCHR$(142)
10  POKE52,48:POKE56,48:CLR
20  POKE56334,PEEK
    (56334)AND254
30  POKE1,PEEK(1)AND251
40  FORI=0TO511:POKEI+
    12288,PEEK(I+53248):NEXT
50  POKE1,PEEK(1)OR4
60  POKE56334,PEEK
    (56334)OR1
70  POKE53272,(PEEK
    (53272)AND240)+12
100  FORI=12296TO12503

```



```

:READA:POKEI,A:NEXT
110 DATA 120,164,36,60,
36,164,66,0
120 DATA 124,162,34,60,
34,162,124,0
130 DATA 24,38,68,64,
64,36,24,0
140 DATA 124,162,34,34,
34,162,124,0
150 DATA 124,162,32,56,
32,162,124,0
160 DATA 124,162,32,56,
32,160,64,0
170 DATA 26,36,64,76,68,
36,24,0
180 DATA 98,164,36,60,
36,164,98,0
190 DATA 56,84,16,16,16,
84,56,0
200 DATA 60,74,8,8,232,
72,48,0
210 DATA 98,162,36,40,
56,164,98,0
220 DATA 96,160,32,32,
34,162,124,0
230 DATA 84,170,42,42,
42,162,68,0
240 DATA 88,164,36,36,
36,164,66,0
250 DATA 48,72,132,132,
132,72,48,0
260 DATA 124,162,34,60,

```

```

32,160,64,0
270 DATA 48,72,132,132,
148,72,52,0
280 DATA 124,62,34,60,
40,164,98,0
290 DATA 126,130,128,124,
2,130,252,0
300 DATA 124,146,16,16,
16,16,56,0
310 DATA 72,164,36,36,
36,36,26,0
320 DATA 68,162,34,34,
20,20,8,0
330 DATA 68,162,34,42,
42,42,20,0
340 DATA 66,164,24,24,
36,66,132,0
350 DATA 68,162,20,8,
8,40,16,0
360 DATA 126,132,8,16,
32,66,252,0

```



54

CARDS SONG

I giochi di carte apparsi su computer non sono, di solito, molto movimentati: per dare uno spirito diverso al programma

sarebbe bene includere dei suoni. Questo programma genera il suono necessario per giocare a carte, cioè smista le carte ai giocatori il secondo.

```

10 REM * * SUONO
PER C64 * *
20 REM * * CARTE SUL
TAVOLO * *
40 FORT=1TO800:NEXTT
50 V=54296:W=54276
:A=54277
60 HF=54273:LF=54272
70 FORY=1TO7
80 POKEV,15:POKEW,
129:POKEA,67
90 POKEHF,70:POKELF,125
100 FORX=1TO600:NEXTX
110 POKEHF,0:POKE
LF,0:POKEW,0
120 NEXTY
130 PRINT"{CLR}":POKE
53280,11:POKE53281,11
:PRINT"{GRAY3}"
250 FORT=1TO800:NEXTT

```



CONOSCI L'INFORMATICA E IL TUO C64/128?

1- L'algoritmo di ordinamento Quicksort ha il numero di confronto eguale a:

- a) $O(n)$
- b) $O(n+1)$
- c) $O(n \cdot \log n)$

2- Una procedura ricorsiva può richiamare se stessa?

- a) No, perchè il computer perderebbe tutti i dati del programma
- b) Sì, a condizione che non vengono passati dei parametri
- c) Sì, sempre

3- Cosa contiene lo stack pointer?

- a) la posizione dell'ultimo valore inserito nello stack
- b) Il valore più alto dello stack
- c) Il puntatore di stack alla memoria dinamica

4- Quale delle seguenti definizioni descrive il sistema UNIX?

- a) Un file system relazionale per la condizione delle risorse del computer
- b) Un sistema operativo multiutente e multitasking per condividere delle risorse
- c) un sistema gerarchico per suddividere più terminali tra i più utenti

5- Quali tra questi algoritmi di ordinamento è il più veloce?

- a) Bubble Sort
- b) Quick Sort
- c) Heap Sort

6- Esiste l'istruzione MOD nel BASIC 2.0 del C64?

- a) Sì
- b) No
- c) Sì, ma solo nel BASIC 7.0

7- Il bus di sistema è un componente necessario alla comunicazione tra:

- a) CPU e sistema operativo
- b) CPU e periferiche di I/O

c) CPU e tastiera

8- la tecnica RISC applicata alla progettazione di microprocessori ha creato:

- a) Delle CPU con poche istruzioni, ma più veloci
- b) Delle CPU senza registri esterni, ma con una potenza di calcolo superiore
- c) Delle CPU più lente ma complesse

9- Uno schema a blocchi permette di descrivere qualunque algoritmo

- a) vero
- b) falso
- c) falso, tranne per algoritmi ricorsivi

10- Nella consultazione di un database relazionale è necessario conoscere...

- a) la posizione di ogni singolo dato
- b) quanti dati ci sono nell'archivio
- c) quali relazioni intercorrono tra i dati

Vedere le risposte a pagina 61

i lettori ci scrivono

WPRO3+ vs EASY SCRIPT

Tra le altre cose, al mio C64 faccio scrivere la corrispondenza o e altri documenti personali utilizzando il programma WPRO3+. Essendo ora venuto in possesso del più potente EASY SCRIPT tutta la corrispondenza la scrivo con questo "nuovo" text editor.

Il mio problema è che posseggo molti file scritti con WPRO3+ e vorrei stamparli e correggerli con EASY SCRIPT, ho provato a caricarli ma niente, potete risolvere il mio problema indicandomi come procedere? Grazie.

L. Taronti - GENOVA

Il programma sotto listato è quello che fa proprio al caso del nostro gentile lettore, basta inserire il nome del file scritto con WPRO3+ (il programma funziona anche con WPRO4) e inserire il nome del file di EASY SCRIPT: la conversione è automatica. Lo provi e ci sappia dire le sue impressioni.

```

100 REM DA WPRO3+ A EASY
    SCRIPT CONVERTER
170 OPEN15,8,15:GOSUB830
180 PRINT"{CLR}{2CUR.GIU}
    {5 CUR.DES}{SHC}
    ONVERTITORE {SH W}
    PRO3+ A {SH E}ASY {SH
    S}CRIPT{CUR.GIU}"
200 INPUT"NOME FILE
    WPRO3+ *{3CURSIN}";F$
210 IF F$="*"THENEND
220 IF LEN(F$)>16THEN
    PRINT"{RVS ON}
    ERRORE!":FORI=1TO500
    :NEXT:RUN
230 OPEN2,8,2,"0:"+F$+"
    ,P,R":IFDSTHEN830
240 PRINT"{2 CUR.GIU}
    OUTPUT EASY SCRIPT"
250 INPUT"{6 SPC}NOME
    FILE -{2 SPC}*
    {3CUR.SIN}";F$
260 IF F$="*"THENEND
270 IF LEN(F$)>16
    THENPRINT"{RVS ON}
    ERRORE!":FORI=1TO500

```

```

280 OPEN3,8,3,"0:"
    +F$+",S,W":IFDSTHEN830
290 GET#2,A1$,A2$
300 GOSUB390:IF (ST)
    AND64THEN370
310 GOSUB830
320 GOSUB470
330 GOSUB650
340 GOSUB570
350 PRINT#3,LN$;
360 GOTO300
370 CLOSE3:CLOSE2:CLOSE15
380 END:RUN
410 LN$="":FORI=1
    TO40:GET#2,A$:
    IFA$="":THENA$=CHR$(0)
    IF (ST)AND64THEN370
420 LN$=LN$+A$
430 NEXT
440 RETURN
450 I=0
490 I=I+1:IFI>40THEN550
500 A=ASC (MID$(LN$,I,1))
510 IFA=31THEN540
520 GOTO500:
530 LN$=LEFT$(LN$,I)
540 L=LEN(LN$)
550 RETURN
560 IFLEFT$(LN$,1)
600 <>CHR$(128) THENRETURN
610 IF MID$(LN$,3,2)
    ="CM" THENLN$=LEFT$
    (LN$,2)+"NB"+MID$(LN$,5)
620 RETURN

```

```

670 FORI=1TOL:F=0
    :A=ASC (MID$(LN$,I,1))
680 IFA= 27 THEN A=91
    :F=1
690 IFA= 29 THEN A=93
    :F=1
700 IFA=124 THEN A=188
    :F=1
710 IFA=126 THEN A=190
    :F=1
720 IFA=122 THEN F=2
730 IFA>63 AND A<91 THEN
    A=A+128:F=1
740 IFA<27THEN A=A+64
    :F=1
750 IFF=1THENGOSUB790
760 IFF=2THENGOSUB810
770 NEXT:IFASC (RIGHT$
    (LN$,1))=31THENLN$=
    LEFT$(LN$,L-1)+CHR$
    (13)
780 RETURN
790 LN$=LEFT$(LN$,I-
    1)+CHR$(A)+MID$
    (LN$,I+1)
800 RETURN
810 LN$=LEFT$(LN$,I-
    1)+CHR$(128)+CHR$(42)+
    MID$(LN$,I+1):L=L+1
820 RETURN
830 REM CONTROLLO DISCO
840 INPUT#15,EN,EM$,ET,ES
850 IFENTHENPRINT
    EN,EM$,ET,ES:STOP
860 RETURN

```





VARIABILI O NO?

Sono un programmatore BASIC alle prime armi, ma ho già constatato che questo linguaggio è molto lento.

Mi è stato detto che per fare dei conti non è consigliabile usare variabili ma i numeri, anche per usi frequenti.

E' vero, o è preferibile ricorrere alle variabili?

S.Stefanini - COMO

Ci spieghiamo con un esempio.

Poniamo in una variabile Y il numero .006347582.

Provate il programma listato, e notate il tempo in jiffies.

```
10 Z=TI:X=1:Y=.006347582
20 FORT=1TO1000
30 X=X+.006347582
40 NEXT:PRINT"X="X:PRINT
   TI-Z"JIFFIES"
```

Ora cambiate la linea 30 con:

```
30 X=X+N
```

e fate ancora partire il programma.

Benché fra i due programmi non ci sia alcuna differenza, la seconda versione è più veloce della prima, così se dovete fare dei conti in un vostro programma, è consigliabile utilizzare variabili.



ELSE?

Mi rivolgo a voi in quanto, essendo un programmatore BASIC che per lavoro opera su un PC con GWBASIC, possiedo anche un C64.

Ho notato però che nel BASIC 2.0 del C64 l'istruzione IF-THEN manca dell'ELSE rispetto al GWBASIC, è possibile simulare tale istruzione?

E.Tircacci - L'AQUILA

Mediante l'istruzione ON X GOTO è possibile simulare l'istruzione IF-THEN-ELSE. Ecco un programmino che mostra come lavora:

```
10 PRINT"ANCORA (S/N) "
20 GET A$:IFA$=""
   GOTO 20
30 ON- ((A$="S")+2*(A$=
   "N"))GOTO 100,200
   :GOTO 10
40 END
```

```
100 PRINT"SI":END
200 PRINT"NO"
```

Se il valore è -1, A\$ è uguale ad S, mentre se il risultato è -2, A\$ è uguale a N. Il meno dopo l'istruzione ON fa divenire i numeri positivi.

Se il valore è 1 il programma va alla linea 100, altrimenti il programma salta alla linea 200. Infine se il valore è 0, non vero, il programma salta ad un'altra istruzione.



LOADING SOUND

E' risaputo che la taratura della testina del registratore Commodore è alquanto critica e quindi difficoltosa. Poiché tale taratura si rende necessaria spesso e volentieri specialmente se si caricano nastri di editori diversi, vengo a chiedere se esiste un sistema per sentire il segnale, un po' come avviene per il Sinclair Spectrum.

Grato per quanto potrete fare, porgo distinti saluti.

F. Fillo - PERUGIA

Se il video di cui usufruisce possiede anche l'audio (non vedo perché non dovrebbe), è possibile ascoltare il segnale del DATASSETTE inserendo POKE 54296,15 ed alzando il volume al massimo. Il livello del segnale non sarà esagerato, ma perfettamente udibile.



TRASFERIMENTO SEQUENZIALE

Oltre al Commodore C64, possiedo anche un PC e a volte mi capita di dover trasferire dei programmi battuti sul C64 sull'altro computer via seriale.

La mia domanda è questa: è possibile registrare i file direttamente in ASCII senza ricorrere ad altri programmi.

S. Scicolone - NAPOLI

Con il comando SAVE è possibile registrare un file in modo sequenziale facendo seguire il nome del file da "S" per cui:

```
SAVE"NOFILE,S",8
```

a questo punto il file viene salvato in maniera sequenziale e non tokenizzato, per richiamarlo bisognerà utilizzare sempre il suffisso ,S.



DISK ERASE

Vecchio possessore di C64, ho da poco acquistato un drive.

La mia domanda potrà essere molto banale ma per me utile: è possibile cancellare tutto un disco senza riformattarlo?

L.Caneparo - BARI

Con la stessa formula della formattazione del disco è possibile cancellarlo interamente avendo l'accortezza di omettere il nome dell'id.

Se vogliamo cambiare anche l'id o il nome senza cancellare tutto bisogna utilizzare invece speciali utilities.



STOP DISABLE

Nella stesura di un mio programma vorrei disabilitare i tasti STOP e RESTORE, è possibile?

F.Gassi - LUCCA

Mediante l'istruzione POKE è possibile abilitare e disabilitare questi tasti, più precisamente con POKE 808,225 avviene la disabilitazione dei due tasti citati mentre con POKE 808,237 avviene la loro riabilitazione.

Sul C128 per disabilitare i tasti STOP e RESTORE bisogna dare POKE 808,103 per riabilitarli bisogna invece dare POKE 808,110.



LIST DISABLE

Volendo tirare uno scherzo ad un amico (che spero non legga la vostra risposta), vorrei trovare il modo di disabilitare il comando LIST sul suo C128? Ringrazio per un vostro gentile riscontro e porgo distinti saluti.

G. Lerione - CATANIA

Visto che lo scherzetto è innocente, eccole il sistema per metterlo in opera.

Per la disabilitazione bisogna usare il solito comando POKE: con POKE 774,138 lo disabilita, con POKE 774,81 lo riabilita.

A scherzo riuscito, mostri pure la risposta al suo amico raccomandandogli di non perdere più neanche un numero di Supercommodore....eviterà altre sorprese.



I libri del mese

LA PROGRAMMAZIONE DEL 6502

Autore: *Rodnay Zaks*
Pagine: 384
Lire: 35.000
Codice: 503B



Questo libro si propone come un testo autosufficiente e completo per insegnare la programmazione del 6502.

Può essere utilizzato da chi in precedenza non ha mai programmato ed è utile anche per coloro che impiegano normalmente il 6502. Alle persone che hanno già programmato, questo libro insegnerà tecniche basate sulle caratteristiche specifiche del microprocessore.

Il testo comprende le tecniche, a livello elementare ed intermedio, richieste per un effettivo inizio alla programmazione e fornisce un vero livello di competenza alle persone che desiderino programmare in l.m.

Naturalmente non vi è libro che insegni effettivamente come programmare se non si eseguono delle applicazioni pratiche.

Il libro porterà il lettore al punto di poter programmare da solo e risolvere problemi semplici, o poco complessi.

Basilare è l'esperienza acquisita dell'autore nell'insegnamento della programmazione dei microcalcolatori a più di 1000 persone.

Il libro, in oltre, è stato progettato per introdurre sistematicamente il lettore a tutti i concetti base e alle tecniche richieste per la costruzione di programmi di complessità crescente.

Sommario:

Concetti di base - Organizzazione hardware del 6502 - Tecniche di programmazione di base - Set di istruzioni del 6502 - Tecniche di indirizzamento - Tecniche di ingresso/uscita - Dispositivi di ingresso/uscita - Esempi di applicazione - Strutture dei dati - Sviluppo del programma - Conclusioni.

COMMODORE 64 IL BASIC

Autore: *Rita Bonelli*
Pagine: 324
Lire: 28000
Codice: 348D



Il libro propone un'accurata esposizione del linguaggio BASIC accompagnata da numerosi esempi: l'unico sistema per impararlo veramente senza troppa fatica.

Un BASIC visto dall'interno pieno di programmi per imparare a programmare.

Nel Capitolo 1 su ha una panoramica dei diversi argomenti.

Il Capitolo 2 è dedicato al linguaggio, il suo indice riporta in ordine alfabetico le parole chiave del BASIC, consentendo una rapida ricerca.

Nel capitolo 3 si forniscono le informazioni necessarie per usare disco e cassetta, per memorizzare programmi. Inoltre si insegna l'overlay dei programmi e si offrono utili suggerimenti per migliorare il proprio lavoro sul computer.

Il capitolo 5 è dedicato alla stampante e insegna a eseguire la copia del video in diversi modi.

Nel capitolo 7 vengono passati in rassegna i codici e i numeri dell'elaboratore.

Il capitolo 8 è dedicato alla memoria, vengono indicate le possibili configurazioni del C64 e sono riportate le tabelle di assegnazione degli indirizzi.

Nel capitolo 9 si trattano gli errori.

Il libro viene completato da due appendici sulla tastiera e sull'argomento del BASIC compilato.

Sommario:

Linguaggio BASIC: costanti, variabili e istruzioni - Memorizzazione e caricamento dei programmi - Stampa su carta - Codici e numeri del calcolatore - Utilizzo della memoria - Appendici.

Progettate i vostri programmi

La programmazione è un hobby eccellente. E' molto divertente far fare al computer ciò che si vuole. Ma, quando si tratta di scrivere un intero programma, bisogna fermarsi un momento e progettare meglio le proprie intenzioni. Come scriverete il programma? Quali passi farete per completare il progetto?

Alcuni programmatori seguono una procedura particolare ogni volta che scrivono un programma. Possono avere imparato i passi da un libro o ad un corso di programmazione, o possono aver preso certe abitudini nell'imparare a realizzare un programma funzionante. Altri programmatori non fanno altro che sedersi e incominciare a digitare. Persino questi programmatori comunque, hanno già in testa molti degli elementi necessari al programma ancor prima di iniziare. Osserveremo varie strategie di programmazione. Se siete dei principianti, scoprirete quali considerazioni sono comuni. Se siete programmatori intermedi, imparerete come affrontare progetti più vasti. Se siete degli esperti, potreste trarre beneficio analizzando le vostre strategie.

Con un'idea precisa in testa

E' qui che molte persone rimangono insabbiate. Non basta un'idea confusa del tipo di programma che volete; è necessario definire con esattezza ciò che dovete realizzare. Molti programmatori di successo riescono a fare questo nella testa, ma può essere d'aiuto scrivere un preciso piano d'attacco. Questo potrebbe include-

**Assumete
il completo controllo
del vostro computer,
programmatelo!
Due veterani
della programmazione
discutono
strumenti e metodi.**

*Rhett Anderson
e
Randy Thompson*

re le specifiche complete di un programma. Disegnare schermi d'esempio e progettare effetti sonori può anche essere d'aiuto.

Spesso vi verrà un'idea quando investigate sulle capacità della macchina. Potrebbe darsi, ad esempio, che mentre state leggendo del VIC II su "La Guida del Programmatore" vi venga in mente una nuova tecnica per lo scroll del video. Poi magari penserete ad un programma che sfrutti questa tecnica. Aggiungete un paio di cose che avete sentito a proposito degli sprite rotanti, ed ecco fatto un programma completo.

Altre volte potete bloccarvi perché conoscete i limiti della macchina. Ascoltare le idee degli amici meno esperti di computer può allargare i vostri orizzonti. I loro pensieri non sono limitati a 8 sprite, 3 voci

sonore, 64 K di RAM e 16 colori.

Il miglior pubblico per le vostre idee siete voi stessi. Se scoprite di aver bisogno di una particolare utility mentre programmate, o se pensate di avere delle buone idee per un word processor, scrivetelo da soli. Se ne avete bisogno voi, può darsi che ne abbiano bisogno anche altre persone. La necessità è la madre dell'invenzione, dopo tutto.

Se tutto il resto fallisce, date una possibilità al subconscio. Molte idee di grandi giochi sono state suggerite dai sogni. Tenete una penna e un blocchetto sul comodino, non si sa mai.

Tenete presente che non tutte le idee sono realizzabili o tantomeno possibili. Quello che può essere un unico ed interessante concetto può rivelarsi impossibile da rendersi effettivo su un microcomputer come il C64.

Spesso, comunque, le idee possono essere ridotte fino ad entrare nei limiti della macchina, e se state progettando un'utility accertatevi che non state creando una soluzione per la quale non ci sono problemi.

La scelta di un linguaggio

Anche se è teoricamente possibile scrivere un programma in quasi tutti i linguaggi, alcuni linguaggi si adattano meglio a particolari tipi di applicazioni (non vi divertireste a scrivere un gioco in COBOL). Per la maggior parte degli usi, il BASIC sarà sufficiente. Potete persino usare un compilatore BASIC come BA-

SIC-64 della Abacus per accelerare il vostro lavoro. Per programmi più avanzati e giochi dove l'azione è veloce e la velocità è critica, il linguaggio macchina è probabilmente la scelta migliore.

I possessori di Commodore 64 e 128 sono fortunati poiché i loro computer possono utilizzare una grande varietà di linguaggi. Linguaggi di programmazione disponibili includono BASIC, linguaggio assembly (o macchina), FORTRAN, COMAL, PROMAL, Pascal, C, Ada e qualche altro. Il BASIC ed il linguaggio assembly sono i più popolari per i computer Commodore a 8-bit per parecchie ragioni. Entrambi i linguaggi sono incorporati nel computer. Se progettate di scrivere un programma da pubblicare in un libro o su una rivista, probabilmente vi conviene usare questi due linguaggi. Potete, ovviamente, usare qualsiasi linguaggio se volete scrivere un programma per vostro uso personale.

Gli strumenti

Le utility per la programmazione rendono la vita più semplice. Programmi di cross-reference, debugger, monitor per il linguaggio macchina, ed estensioni BASIC che aggiungono istruzioni utili di editing, hanno tutte la loro importanza. Con il giusto strumento, potete dimezzare il tempo della programmazione.

Alcune utility possono essere usate con parecchi linguaggi. SYRES (della Solidus International) per esempio, può essere usato per editare il BASIC 64 o file di testo in linguaggio assembly. Editor di sprite, di caratteri e di suono, sono tutti strumenti vitali per la programmazione. La nostra rivista ha pubblicato parecchi editor di questo tipo.

Se scegliete di scrivere il vostro programma in BASIC, potreste prendere in considerazione la possibilità di acquistare una utility di cross-reference come X-ref 64 della Abacus. Questo tipo di programmi valuta il vostro programma e crea stampe che elencano variabili di programmi e reference su numeri di linea, un inestimabile strumento per il debugging. Potreste anche considerare l'uso di un intensificatore di editor BASIC come Syres.

Queste utility offrono istruzioni di traccia che vi consentono di tracciare flussi di

programma.

Se invece usate un altro linguaggio, come il Pascal o il C, potete usare gli strumenti forniti nella confezione.

Decisioni, Decisioni

Una volta che la vostra idea è ben chiara e le condizioni "ambientali" per la programmazione sono ben "sintonizzate", è tempo di iniziare a fare delle scelte difficili. Che vi piaccia o no, ci sono certe restrizioni a cui ogni programmatore deve sottostare.

Queste includono il tempo, le limitazioni dell'hardware e quelle della vostra abilità ed esperienza. Se provate ad intraprendere un lavoro troppo difficile o troppo presto nella vostra carriera di programmatore, siete destinato a rimanerne frustrato. Fate un passo indietro e provate qualcosa di meno velleitario.

**Prendete
qualche spunto
dagli altri media.
La prossima volta
che vi capiterà
di guardare
Tron o Guerre Stellari,
o alla prima
occasione che avrete
di ascoltare
il notiziario notturno,
provate a prendere
qualche idea
originale
per un vostro
programma.**

Quale sarà l'aspetto e l'atmosfera del programma? Necessiterà la grafica? Se sì, di che tipo? Userete gli sprite? Lo schermo ad alta risoluzione? Caratteri ridefiniti? Documentatevi sulle tecniche relative alla grafica della quale avete bisogno.

E l'interfaccia utente? Se il programma è un gioco arcade avrete probabilmente intenzione di usare un joystick. Un gioco di strategia potrebbe sfruttare sia la tastiera che il joystick. Se invece state progettando un'applicazione di produttività, converrà considerare mouse, menu a pull-

down e finestre. Uno sforzo maggiore qui, renderà il programma più facile da usare e molto più piacevole alla vista.

Per scoprire se il vostro programma è intuibile, fatelo provare ad un utente di computer inesperto. Le persone che non hanno dimestichezza fanno cose che voi non potreste mai pensare. E' importante che tutti siano capaci di usare il programma. Per riuscire a realizzare ciò, potreste aggiungere messaggi d'errore esplicativi o schermi di aiuto. Uno schermo d'aiuto ben scritto può sostituire pagine di documentazione.

Organizzazione

Quando siete pronti per scrivere il programma, dovete decidere con esattezza quale sarà la sua organizzazione. Molti programmatori scindono il loro codice in subroutine o subprogrammi, dove ogni sottosezione adempie ad un compito specifico. Tali sezioni possono venire provate indipendentemente, prima di essere unite insieme in un programma completo. E' molto importante il modo in cui strutturate ogni sezione. Alcune parti potrebbero necessitare il linguaggio macchina per un'alta velocità. Altre potrebbero essere interrupt-driven, come la musica di sottofondo o l'animazione degli sprite.

L'organizzazione della memoria può essere critica nei programmi del C64, in particolare modo per quelli che fanno uso di schermi ad alta risoluzione, sprite o caratteri ridefiniti. Il blocco di memoria da 4K, che ha inizio alla locazione \$C000 (49152) viene spesso usato per l'immagazzinamento dei dati. Se il vostro programma deve interagire con un'utility, tenete presente che molte utility usano la totalità o parte di questo spazio. Se la vostra organizzazione della memoria sarà un po' complicata o inusuale, disegnatene una mappa della memoria che mostri dove verrà situato ogni elemento che necessita di spazio.

In questi ultimi anni, sono stati insegnati molti metodi per l'organizzazione della memoria. I diagrammi di flusso sono stati in voga per qualche tempo. Un diagramma a flusso è un grafico della logica del programma. E' un'immagine ad alto livello di quali saranno le decisioni e fun-

zioni dei tasti del programma.

Molti programmatori fanno uso dello pseudocode quando progettano i propri programmi. Lo pseudocode è simile ai linguaggi procedurali come il Pascal ed il C, ma è integrato da frasi in italiano corrente.

Ora bisogna scegliere tra programmazione modulare, top-down, e bottom-up. Nella programmazione top-down si progetta prima il flusso generale del programma con un diagramma di flusso o con lo pseudocode.

Dopo aver tracciato il programma principale, si tracciano le subroutine delle quali questo farà uso. Queste subroutine potranno, a loro volta, appoggiarsi ad altre subroutine. Prima o poi raggiungerete il fondo di questa progressiva diminuzione di grandezze e dovrete occuparvi dei dettagli.

Nella programmazione bottom-up, si iniziano a scrivere le routine ritenute necessarie al programma al quale si lavora. Queste routine possono essere manovratrici di sprite o suoni a basso livello. Iniziando da questa routine, si costruisce un programma che connette i pezzi.

La programmazione modulare unisce la programmazione top-down e bottom-up. Nella programmazione modulare, si definiscono le interfacce per le routine necessarie. Ogni modulo può venir provato indipendentemente, prima di essere usato.

Il prodotto finale

Prima o poi stabilirete che il programma è finito. Dategli un'ultima occhiata. Avrete probabilmente risolto alcuni problemi durante questa dura prova. Tenete a mente le soluzioni (e, ancor più importante, i metodi che avete usato per trovare le soluzioni) per il prossimo programma che scriverete.

Se il programma è destinato a un vostro uso personale, avete finito. Se è destinato ad un amico o alla vendita, questo è il momento di fare un beta-test del programma e di scriverne la documentazione. Visto che avete scritto voi il programma, conoscerete bene come funziona ogni caratteristica. Se questo non sarà ovvio anche per le persone che proveranno il

software, dovrete apportare alcune modifiche al programma o essere più espliciti nelle istruzioni. La documentazione è una parte importante di ogni programma.

Terminare il vostro primo programma sarà un'esperienza eccitante. E con ogni nuovo programma che scriverete, troverete strumenti e tecniche personali per affrontare nuovi e più complicati problemi. Nell'articolo seguente vedrete come si affronta praticamente un problema di programmazione.

**Se vi accingete
a scrivere un programma
di una certa
complessità, dovrete
preferibilmente optare
per uno dei metodi
qui descritti.
Nel caso in cui aveste
la sensazione
di perdervi,
provate a scoprire
nuove strade
per accostarvi
in maniera diversa.
Regola principale
è quella di non
demordere mai,
se non insistete, non
riuscirete a raggiungere
lo scopo che vi siete
prefissato e questo
farebbe inevitabilmente
di voi un programmatore
di seconda serie.**

Glossario dei termini

ASSEMBLER

Un programma che traduce il codice sorgente in linguaggio assembly, quindi in linguaggio macchina. E' essenziale per ottenere una certa velocità.

BETA-TEST

Il processo della verifica di un programma, eseguito da altri oltre che dal programmatore che ha realizzato il programma stesso.

COMPILATORE

Un programma che traduce il codice sorgente di un linguaggio ad alto livello in

linguaggio macchina.

DEBUGGER

Un ambiente che vi permette di interagire con il codice oggetto del programma. Caratteristiche tipiche sono single-stepping delle istruzioni di un programma ed analisi delle variabili ed è possibile rilevare inesattezze nella programmazione.

DOS WEDGE

Un programma abbinato al computer e comunemente usato anche per il C64 e il C128. Esso consente un facile accesso al disk drive.

DIAGRAMMA DI FLUSSO

Una rappresentazione grafica del flusso logico di un programma.

LINGUAGGIO AD ALTO LIVELLO

Un linguaggio "portatile", come il BASIC o il Pascal, che fornisce astrazioni che consentono al programmatore di ignorare i dettagli del codice macchina.

INTERRUPT-DRIVEN

Descrive una sezione del codice del programma che opera un'interruzione del microprocessore.

MONITOR

(IN LINGUAGGIO MACCHINA)

Un programma che vi permette di lavorare interattivamente con il codice linguaggio macchina e dialogare quindi direttamente con la CPU.

MAPPA DELLA MEMORIA

Un diagramma che descrive graficamente l'organizzazione della memoria usata da un programma e illustra le zone di memoria occupate dal sistema.

CODICE OGGETTO

Il codice generato da un compilatore di linguaggio ad alto livello o da un assembler.

PSEUDOCODICE

Una sezione di codice scritto in una combinazione di un linguaggio ad alto livello e del linguaggio umano.

CARATTERI RIDEFINITI

Definizione sostitutiva per uno o più caratteri. Spesso usati per caratteri di lingue straniere o caratteri grafici sullo schermo di testo standard. I caratteri vengono ridefiniti dall'utente in funzione della natura del programma che sta scrivendo.

CODICE SORGENTE

Codice scritto in un linguaggio ad alto livello o in linguaggio assembly.

I problemi della programmazione

Scrivere un programma, come qualsiasi altro compito, è molto più semplice se il problema principale viene frammentato per essere meglio affrontato.

Iniziate con un quadro generale dei compiti che volete vedere svolti dal vostro programma e poi dividetelo in parti più piccole e meno complesse. Vedremo come funziona questo modo di procedere scrivendo un programma d'esempio sul Commodore 64.

Nello svolgimento, possiamo esaminare alcuni importanti aspetti della programmazione.

L'idea

Il primo passo per scrivere un programma è quello di decidere quali compiti desiderate che il programma porti a termine. A fini dimostrativi, scriveremo un piccolo gioco in BASIC.

Un classico relativamente facile da scrivere è il gioco "dell'intrappolamento". In questo gioco, due giocatori sono inizialmente posizionati nel mezzo di un campo da gioco circondato da un bordo. Quando il gioco ha inizio i due giocatori avanzano l'uno verso l'altro e ognuno lascia una traccia dietro di sé. Lo scopo del gioco è quello di evitare tutti gli ostacoli, la vostra stessa traccia, la traccia dell'avversario, ed il bordo per più tempo possibile. Ora che sappiamo di cosa tratterà il programma, possiamo organizzare i nostri strumenti per affrontare il lavoro. Il programma è scritto in BASIC e forse vi saranno utili alcuni manuali di riferimento. Due eccellenti fonti dalle quali si possono at-

Patrick Parrish

tingere numerose tecniche di programmazione, sono "La Guida di riferimento del programmatore" ed il manuale allegato alla macchina.

Una struttura procedurale

Prima di iniziare a scrivere il programma, dobbiamo tracciarne uno schema.

Dobbiamo cioè elencare in ordine, le principali routine che compariranno nel programma.

Ricordatevi che nello sviluppo del programma stesso alcune routine verranno cancellate, ed altre aggiunte. Questo è assolutamente naturale, quindi non preoccupatevi se non tutto è a posto sin dal primo momento. La vostra lista procedurale potrà e DOVRA' cambiare lungo il cammino.

Una procedura ideale per il nostro programma potrebbe essere:

1. Inizializzare le variabili.
2. Organizzare il campo di gioco.
3. Ultimo round? se sì, allora vai al punto 15.
4. Controllo joystick giocatore 1.
5. Aggiorna posizione giocatore 1.
6. Collisione del giocatore 1?
7. Se sì, aggiornare punteggio e numero del round; poi vai al punto 2.
8. Se non ci sono collisioni

muovi giocatore 1.

9. Controllo joystick giocatore 2.
10. Aggiorna posizione giocatore 2.
11. Collisione del giocatore 2?
12. Se sì, aggiornare punteggio e numero del round; poi vai al punto 2.
13. Se non ci sono collisioni muovi giocatore 2.
14. Vai al punto 4.
15. Routine di fine gioco.

Se sapete costruire un diagramma di flusso, forse preferite rappresentare così lo schema procedurale. Dopo aver fatto la lista di tutte le routine del programma, il progetto si semplifica molto. Se non sapete bene come scrivere ogni routine, cercate nel vostro Reference Book routine simili in altri programmi. Anche riviste o libri con listati di programmi possono essere utili.

Scrittura del programma

Visto che si tratta di un gioco, la routine più importante rispetto a tutto lo sviluppo del programma è la seconda; organizzare il campo di gioco. Incominciamo.

Prima, usando una mappa della memoria dello schermo (se ne trova una nella vostra guida utenti) disegnate il campo di gioco come l'avete immaginato. Il campo di gioco è circondato da un bordo, fatta eccezione per la prima e l'ultima linea dello schermo, che contengono il punteggio di ogni giocatore.

(segue a pagina 19)

Guida all'acquisto degli

Strumenti per programmare

Caroline D. Hanlon

Siete interessati seriamente alla programmazione? Se sì, potete rendervi la vita molto più facile scegliendo il giusto strumento. Questa guida elenca cosa c'è di meglio sul mercato per C64 e C128.

ADA Training Course

Abacus

Ada è il linguaggio per lo sviluppo dei programmi del Ministero della Difesa Americana, e il prodotto in questione contiene tutto il necessario per programmare in tale linguaggio. Il sistema comprende un editor, un analizzatore sintattico ed un compilatore, un assembler, disassembler e manuale

Assembler/Monitor

Abacus

Consente la programmazione in linguaggio macchina con il C64. L'assembler possiede capacità di macro assembler, assembly condizionale, editing a pieno schermo, lista della tavola-simboli e linking di file-sorgente. Può assemblare su disco, nastro o in memoria. Il monitor può cercare, disassemblare il codice, trasferire o comparare blocchi di data, avere accesso ad altre banche di memoria, quick trace con il breakpoint o effettuare esecuzione passo per passo. Monitor ed assembler possono coesistere e sono entrambi scritti in linguaggio macchina.

BASIC-128

Abacus

Commodore 128.

BASIC-128 è un compilatore atto ad accelerare i programmi in BASIC 2.0 e

7.0 sul 128. Il programma è utile per l'amministrazione della memoria, per compilare programmi in speed-code (pseudocode) o linguaggio macchina ottimizzare loop FOR-NEXT, inviare istruzioni al disk drive o cambiare parametri di compilazione. La versione 128 offre funzioni matematiche ad alta precisione e usa tecniche d'ottimizzazione di numeri interi e formule. La confezione contiene anche un modulo runtime.

BASIC-64

Abacus

BASIC-64 è un compilatore per incrementare la velocità dei programmi BASIC. Tra le caratteristiche: amministrazione della memoria, compilazione dei programmi in speed code (pseudocode) o linguaggio macchina, ottimizzazione cicli FOR-NEXT, invio istruzioni al disk drive, cambio dei parametri di compilazione.

La versione 64 può compilare i programmi in BASIC 2.0 sfruttando l'opzione overlay ed estensioni BASIC quali Simon BASIC, VICTREE, e BASIC 4.0. Modulo runtime incluso.

COBOL-64 e 128

Abacus

Commodore 64 e 128.

COBOL è stato progettato per aiutare gli utenti nella programmazione in

COBOL con C64 e C128. Propone tra le caratteristiche: editor con analizzatore di sintassi, compilatore, interprete, funzione crunch per ridurre la dimensione della memoria, programmi d'esempio, e strumenti di debugging simbolici come breakpoint, trace, single step. Include un sottinsieme del ANSI COBOL 74 e ne esistono versioni per il C64 ed il C128. La versione 128 funziona nel modo a 40 o 80 colonne. Manuale di 150 pagine.

Commodore LOGO

Terrapin

Commodore logo è un ambiente attivo d'apprendimento, utile per insegnare ai bambini il ragionamento matematico, risoluzione dei problemi e creatività. L'opzione di elaborazione liste può anche essere usata con arti del linguaggio. Il programma combina testo ed immagini e usa turtles (la tartaruga) per disegnare figure e realizzare animazioni. Questa introduzione alla programmazione del computer contiene 7 sprites, 15 colori e musica avanzata. Prodotti compatibili, sempre della Terrapin, comprendono: materiali di supporto al corso per uso scolastico Logo works e prodotti Logoware come per esempio il Logo Data Toolkit. Commodore Logo è anche disponibile in confezione da 5 dischi e da 10 dischi.

The Final Cartridge III

Datel Electronics

Si tratta di un sistema operativo basato su cartuccia per C64 e C128. Presenta menu a pull-down, finestre e più di 60 istruzioni e funzioni. Il calcolatore riceve input da tastiera, mouse e joystick e rende possibile l'uso del tastierino numerico del 128 nel 64 mode. Note pad è un mini word-processor con caratteri proporzionali. Un menu freezer può essere usato per stampa e screen dump, eliminare sprite, mettere il joystick in auto-fuoco, proteggere il computer quando si cambiano le porte joystick, e fare il back up su nastro o disco. Il monitor LM non risiede nella memoria; contiene uno sprite editor, un character editor, un gestore di drive, driver per stampante e funzioni di scorrimento.

FORTH Language

Abacus

Questo prodotto contiene un vocabolario esteso che prevede grafica e suono, un editor a pieno schermo incorporato, un assembler FORTH, ed un libretto. All'occorrenza possono essere create parole nuove.

GeoProgrammer 2.0

Berkeley Softworks

GeoProgrammer 2.0 è un programma di sviluppo per il linguaggio assembly che consente agli utenti di realizzare applicazioni GEOS a piena scala. GeoProgrammer utilizza icone, menu ed interfacce a finestra. Il programma comprende GeoAssembler, GeoLinker, e GeoDebugger. Può venire usato con GEOS 64 e GEOS 128.

JiffyDOS

Creative Micro Designs

Commodore 64 o 128.

JiffyDOS è una sostituzione ROM del bus seriale per la CPU e anche per il disk drive. E' disponibile per tutti i modelli di C64 e C128, come anche per la maggior parte dei disk drive. Qualsiasi ROM di computer funziona con qualsiasi ROM di drive e combinazioni di drive, e le CPU e disk drive possono venir scambiate e abbinate liberamente. Per i programmi fortemente protetti, c'è un interruttore sul computer e drive. I tasti funzione pre-programmati sono progettati per funzionare congiuntamente ad una direzione sullo schermo.

Sono stati aggiunti 14 nuovi comandi alla sintassi standard del DOS Wedge, e tutti possono venire usati sia nel program che nel direct mode. Jiffy-

DOS serve anche per accelerare i file MIDI. E' semplice da installare e include manuale e garanzia "soddisfatti o rimborsati". Sono anche disponibili ROM drive extra.

Kyan PASCAL/128

Kyan Software

Commodore 128.

Questo programma è un'implementazione dell'ISO Pascal, con menu di istruzioni, schermi di help, ed un manuale da 300 pagine e Pascal tutorial. Fra le altre caratteristiche: un assembler di LM 6502, un editor di testo a pieno schermo, estensioni Pascal, un macro assembler incorporato, e dischi non protetti.

Kyan PASCAL/64

Kyan Software

Kyan Pascal/64 un'implementazione del Jensen/Wirth Pascal.

Propone un compilatore standard avanzato su ogni disco in modo da venire incontro ai programmatori di tutti i livelli. Include un editor di testo a pieno schermo, un sistema di amministrazione dei file, un ambiente runtime a sé stante, e estensioni per la grafica. La versione avanzata mantiene queste caratteristiche e ne presenta di nuove, un compilatore LM, un assembler incorporato, estensioni Pascal, e file intermedi in linguaggio assembly.

Linear Programmer

Computer heroes

Serve per risolvere problemi lineari di programmazione con un numero massimo di 50 variabili e 35 costanti, incluse equazioni e disequazioni. La funzione "Z" può essere massimizzata o minimizzata. La soluzione viene mostrata sullo schermo.

MAE 64 Assembler, 5.0

Schnedler Systems

Questo macro assembler 6502 e 65C02 offre uno screen editor co-residente con BASIC Micromon. L'assembler presenta assemblamento condizionale, interattivo, pseudo-ops, ed error-checking. E' incluso un manuale di 89 pagine.

MASTER-64 Development Tools

Abacus

Master-64 è un prodotto di vasta portata per lo sviluppo delle applicazioni: gestione dello schermo, direzione di file a indice, matematica a multiprecisione, ed un monitor LM. Può aggiungere fino a 100 comandi al BASIC. Le caratteristiche includono sistema di

file ISAM, generazione stampante, estensioni BASIC, sussidi, istruzioni di BASIC 4.0, ed un monitor LM.

Menu-Driven Operating System

Cardinal Software

Con questo programma, le istruzioni che riguardano i dischi possono venire eseguite leggendo il menu e premendo un tasto, ed il linguaggio macchina viene convertito in BASIC. Istruzioni ad un tasto includono operatori di programmi come LOAD e SAVE ed istruzioni come copy, scratch, rename, format, renumber, e combine.

Merlin 128

Roger Wagner Publishing

Commodore 128.

Merlin 128 è un macro assembler specifico per il 128, che assembla a/disco e prevede assembly condizionale. Converte numeri interi in numeri a virgola mobile, linkers per generare codice rilocabile. I numeri di linea appaiono quando il codice sorgente viene listato con il numero della linea attuale nel modo edit. Gli utenti possono anche scorrere durante l'editing. Il programma funziona solo nel modo a 80 colonne. Merlin 128 include programmi dimostrativi delle macro e sourceror, un disassembler. Circa 35K di codice sorgente possono venire inseriti nella memoria in una volta.

Micro Detective Automatic Error Detection

American Made Software

Commodore 64 o 128.

Questo debugger per il BASIC del Commodore 64 o 128, offre individuazione automatica di errori e possibilità di tracing. Può trovare e motivare molti errori, come quelli di sintassi e di illegal quantity. La routine di "bug-buster trace" mostra l'esecuzione di ogni linea e le variabili coinvolte. E' anche possibile elencare le variabili, file e loop usati nel programma. La funzione trace può essere attivata o disattivata a piacere. La versione C128 è compatibile con BASIC 7.0 e finestre. Una finestra trace può venir messa dovunque su uno schermo a 40 od 80 colonne. Ogni confezione contiene un manuale di istruzione dettagliato.

PASCAL-64

Abacus

Pascal 64 è un compilatore Pascal a livello introduttivo progettato per il C64. I programmi si compilano nel LM 6510.

PTD-6510 Symbolic Debugger, 4.0 *Schnedler Systems*

Questo debugger trasparente simbolico funziona a finestre che mostrano le variabili durante procedimenti passo per passo, un mini-assembler a two-pass e patching automatico. E' anche compatibile con programmi driver ad alta velocità per stopping automatico, condizionale e mostra i passi più importanti del C128. Per programmi in linguaggio assembly 6502. Il manuale è composto da 100 pagine ed è incluso nella confezione.

Quick Brown Box *Brown Boxes*

Quick Brown Box è una cartuccia che contiene RAM a batterie da 16K, 32K o 64K che consente all'utente di creare cartucce personalizzate di programmi BASIC e LM. La versione da 32K contiene fino a 128 blocchi; la versione 64K ne contiene 255 e la directory della cartuccia può tenere 30 entrate. Con l'opzione Ramdisk, gli aggiornamenti possono venir salvati direttamente su Quick Brown Box. La cartuccia può venir divisa in settori per applicazioni speciali ed eseguire l'autoboot al disk drive. I programmi manager governano la creazione e l'alterazione dei contenuti della Box. Un interruttore permette la selezione del mode 64 o 128. La Box include Supermon+ ed un programma di checksum. La batteria interna al litio da 3 volt, ha una vita media di 10 anni. Un disco separato di utilities contiene manager, istruzioni di UNNEW e KILL, routine, screen dump, ed è anche disponibile un programma terminale.

Super C *Abacus*

Commodore 64 o 128.
Super C funziona con il C64 o il C128 e serve per produrre codice macchina 6502. Questo sistema di sviluppo del linguaggio C include: un editor, un compilatore un linker e librerie grafiche e matematiche.

Il codice sorgente può contenere linee da 80 caratteri e scorrimento orizzontale. I programmi sorgenti possono essere lunghi fino a 41K, mentre il codice oggetto può essere di 53K. Fino a 7 moduli possono essere combinati con il linker, e la libreria runtime può essere richiamata dal linguaggio macchina o essere inclusa come programma tipo BASIC. La versione per il C128 è compatibile con i Ramdisk.

Super PASCAL-64 o -128 *Abacus*

Commodore 64 o 128.
Super Pascal è un sistema di sviluppo disponibile nella versione per il Commodore 64 ed il 128. Include: editor di file-sorgente, assembler, pacchetti runtime e utility, più caratteristiche come aritmetica da 11 cifre ad alta precisione, overlay, caricamento automatico di programma sorgente ed editor, messaggi d'errore, localizzazione durante la compilazione e rapporti statistici. Il sistema usa il compilatore Jensen e Wirth ed estensioni per la grafica. La versione per il C128 presenta un pacchetto grafico da 80 colonne e alta risoluzione, burst mode per il 1571 ed un Ramdisk.

Symbolic Master Multi-Pass **Symbolic Disassembler** *Schnedler Systems*

Questo disassembler simbolico può convertire 6502, 6510, opcode non documentato, programmi in LM 65C2 e 8502 in codice sorgente. Etichette codificate sono usate per mostrare la struttura e il flusso del programma. Si può fare l'output su disco dei file di codice sorgente. Manuale da 63 pagine compreso.

Video BASIC-64 *Abacus*

Video BASIC-64 incrementa il BASIC con 50 istruzioni relative alla grafica e agli effetti sonori. Offre runtime e supporto per stampanti a dot-matrix.

XREF BASIC Cross Reference *Abacus*

Commodore 64 o 128.
XREF BASIC Cross Reference può essere usato per fare il cross reference delle variabili, dei numeri di linea, costanti numeriche e parole chiave BASIC per il debugging dei programmi BASIC. La cross reference è realizzata e listata su schermo o stampante. Parole chiave non-Commodore possono essere cross-referenced. I programmi possono venire letti dal disco. Sono disponibili versioni separate per C64 e C128.

XYTEC Macro Set 1 *Xytec*

Xytec Macro Set 1 è una collezione di 58 macro per accelerare la programmazione. La confezione offre più di 40 subroutine, più di 4000 linee di codice e sussidi per il debugging. Funziona con dischi compatibili DOS e modifiche come il 1541 Flash e il JiffyDOS.

(segue da pagina 16)

I caratteri che rappresentano i due giocatori sono posizionati sullo schermo insieme ad un messaggio che dice agli utenti come iniziare il gioco.

Poi, usando il diagramma dello schermo, scrivete la routine che disegna il campo da gioco (linee 140-260). Notate che abbiamo fatto il POKE del byte del colore corrispondente prima di fare il POKE di ogni carattere del bordo. Questo eviterà un effetto di tremolio, se il colore di sfondo dovesse essere diverso dal colore del quale abbiamo fatto il POKE. Inoltre, abbiamo definito il carattere e colore del bordo, come anche quelli dei giocatori, come variabili. Potreste cambiarli più avanti con un pò di sforzo, magari per dare un look diverso al gioco.

Il resto del programma si sviluppa da questa routine. Il listato è liberamente commentato con vari REM, non dovrete quindi avere problemi nel seguirlo.

Migliorie

Una volta che il programma sta funzionando, vi potete concentrare nel debugging e nel migliorarlo.

Ci sono parecchie migliorie che potete apportare. Tra queste vi è il suono, i caratteri ridefiniti per i giocatori e varie opzioni di gioco, (per esempio fornire diverse velocità e posizionare casualmente alcuni ostacoli nel campo). Nel caso il gioco vi sembri un pò lento, potete renderlo più veloce in diversi modi.

Primo, potete definire tutte le costanti nel loop del gioco (linee 300-480) come variabili. Ma per perfezionarlo al massimo, dovrete riscrivere il loop in linguaggio macchina (LM).

Se avete poca dimestichezza con il linguaggio macchina, un breve programma come questo è l'ideale per acquisire un pò di esperienza. Iniziate con routine semplici, piuttosto che provare a scrivere un intero programma. Ovviamente, per intraprendere questo progetto, avete bisogno di un assembler (PAL della Spinnaker Software è superbo) ed un monitor LM. Inoltre, qualche libro sul LM può sempre far comodo.

Vedere il listato a pagina 60.

Jackson



SOFTWARE & RIVISTE

D.O.C.

Scegli i giochi più freschi, quelli appena sfornati dal mercato internazionale. Poi, in edicola, chiedi le riviste più ricche di informazioni e idee per utilizzare il tuo computer nel modo più O.K. Ora unisci al tutto qualità, esperienza, tanta serietà, una lunga tradizione di fiducia e un pizzico di divertimento. Ecco fatto. Hai trovato il marchio di qualità che solo Jackson ti può offrire: è il marchio Jackson Software D.O.C. D'ora in poi cercalo sempre.



**GRUPPO EDITORIALE
JACKSON**

AREA CONSUMER

Scegli il meglio: scegli Jackson.

Compila e spedisce il coupon a: **GRUPPO EDITORIALE JACKSON AREA CONSUMER**
via Rosellini, 12 - 20123 MILANO
Riceverai gratuitamente il nuovissimo adesivo "Jackson D.O.C."



NOME _____ COGNOME _____

INDIRIZZO _____ TEL. () _____

CAP. _____ CITTÀ _____ PROV. _____

ETÀ _____ PROFESSIONE _____

QUALE COMPUTER POSSIEDI? _____

QUALI RIVISTE LEGGI? DI INFORMATICA _____

ALTRE _____

SEI ABBONATO AD UNA RIVISTA JACKSON? SI ☐ NO ☐

QUALE? _____

SEI TITOLARE JACKSON CARD? SI ☐ NO ☐

SILVER CARD N° _____ GOLD CARD N° _____

SuperCommodore Clips

a cura della Redazione
e, dagli U.S.A., Mc Lean (Compute!)

NEWS, CURIOSITA' E NUOVI PRODOTTI

Jordan vs Bird

Un'altro gioco di basket, ma questa volta dalla Electronic Arts.

Il gioco vede le due superstar del basket NBA uno contro l'altro, ma non solo, ci sono gare con tiri da tre punti e gare con schiacciate.

I due giocatori nella realizzazione del gioco hanno assistito i disegnatori e programmatori per offrire a questi il loro modo di giocare e il loro carattere.

disegnare diversi tipi di grafici da quelli a barre a quelli a torta.

Le caratteristiche di questo pacchetto includono la manipolazione di caratteri, font, grandezza dei font, grandezze dei punti e tipi di stile. Indispensabile per visualizzare in 3D l'andamento di varie grandezze a confronto.

Michael Jordan spiega i punti cruciali della sua gara per Electronic Arts.



Diagrammi con Geos

La Berkley Softworks ha recentemente realizzato un nuovo programma per il sistema operativo GEOS.

Il nome del programma è GeoChart, gli utenti possono disegnare i propri caratteri con dati importati da GEOS 1.2, 1.3 e 2.0; GEOS 128; GeoCalc 128.

Con questo nuovo pacchetto è possibile

Supreme Challenge

La Firebird ha messo assieme un pacchetto formato da cinque tra i più affermati game, prendono parte: The Sentinel, Star-glider, Elite, Ace 2 e Tetris. La versione della compilation è stata realizzata, oltre che per il Commodore 64, anche per Amstrad e Spectrum. Visto il peso dei game, il successo è assicurato.

Brainstick

Chi ha visto il film "Firefox" di Thomas Craig, sicuramente si ricorderà del modo in cui Clint Eastwood, protagonista principale nonché pilota del futuristico aereo, riusciva a pilotarlo. Non usava infatti gli arti convenzionali bensì la sola forza del pensiero.

La risposta di controllo più veloce e precisa che il Cerebral Interfacing System offre, è stata ed è tutt'oggi soggetto di intensive ricerche da parte della NASA e dell'USAF.

La Home Cyberlabs, una compagnia britannica con sede all'isola di Wight, ha percorso i tempi realizzando una piccola interfaccia che come input ha dei sensori da applicare alle tempie e come output un connettore Cannon DB9, come quello dei joystick convenzionali, da inserire appunto nella relativa porta dell'home computer. Il Brainstick, così è stata chiamata questa interfaccia, rileva le onde alfa emesse dal cervello, le decodifica e le trasforma nelle informazioni relative agli otto movimenti più lo sparo.

Ma, non è che il sistema richieda livelli di concentrazione eccessivi da parte del giocatore? "Non è così", assicura Joseph King della Home Cyberlabs, "abbiamo realizzato l'interfaccia elaborando una

Il pacchetto formato da cinque supergame studiato dalla Firebird.



circuiteria appropriata contenente un algoritmo di controllo rivoluzionario e degli speciali filtri che assicurano in ogni momento la massima affidabilità. Le nostre ricerche hanno rivelato che l'intensità dello sforzo mentale richiesto da Brainstick, non raggiunge quella di solito in gioco usando i comuni joystick con il vantaggio che il processo cervello-azione è diretto".

Il più grosso ostacolo al perfetto funzionamento di questa interfaccia era l'ampia variazione delle onde alfa da un individuo all'altro, ma tale handicap è stato scavalcato facendo richiedere al sistema una sequenza di startup di una decina di secondi durante i quali esso calibra il circuito di decodifica adattandolo alla fisiologia del soggetto e registrandone i parametri in un banco di RAM alimentato da una piccola batteria. Fatto ciò basta inserire il relativo spinotto in porta joystick e caricare il game preferito.

Probabilmente Brainstick avrà un certo successo specialmente per quei giochi tipo corsa in macchina come Wec le Mans, Out Run o come Thunderblade e Operation Wolf, ma molto più difficilmente potrà essere impiegato ad esempio nella simulazione dei giochi olimpici come Daley Thompson's Olympic Challenge che richiedono rapidi movimenti destra-sinistra difficili da riprodurre mentalmente.

Navigator

Già qualche numero fa avemmo modo di parlare del Navigator, uno dei joystick più all'avanguardia oggi giorno. Torniamo un attimo sull'argomento presentando la struttura interna di questo particolare joystick che sembra voler prendere decisamente piede anche in Italia.

A parte la forma ergonomica dell'impugnatura che ne permette un comodo uso sia per destri che per i mancini, c'è da segnalare il particolare sistema con il quale la cloche, la cui parte esterna sporge per non più di 2 cm dal telaio, aziona i quattro sensibilissimi microswitch posti su di un piano fissato al disopra del fulcro della cloche stessa. Sullo stesso piano trova posto anche il commutatore di autofire, mentre il pulsante di sparo presenta all'esterno un comodo grilletto che la sensazione dello sparo la dà davvero.

Il Navigator presenta un look futuristico e una comoda impugnatura.



Con Chord Printer diventare chitarristi è facile. Gli accordi non saranno più tabù.

Tutti chitarristi

Agli insegnanti e agli studenti di chitarra che posseggono il 64, interesserà sicuramente Chord Printer, il nuovo programma della The Guitar Connection con il quale tutti hanno la possibilità di divenire chitarristi.

Chord Printer è un programma basato su un menu, comprensivo di un dizionario che spiega la posizione delle dita sulle corde per formare maggiori, minori, settime sia maggiori che minori, seste sia maggiori che minori, diminuite, aumentate ed altre combinazioni.

Chord Print mostra ad una ad una le dita selezionate per formare l'accordo e ne permette la stampa su carta e diagrammi.



European Computer Trade Show

Si è tenuto in Inghilterra dal 16 al 18 aprile '89, il primo salone micro che raggruppa editori inglesi, italiani, tedeschi, spagnoli, francesi e anche americani.

Concorrente dell'americano Consumer Electronic Show, fino ad oggi luogo d'incontro privilegiato dei personaggi europei della microinformatica, l'European Computer Trade Show (ECTS) fa il suo ingresso nello scenario europeo gettando subito solide basi.

Vengono affrontati problemi comuni a tutti come il sistema di tradurre i programmi, il sondaggio dei vari mercati o, ancora, la vendita per corrispondenza. L'ECTS tenterà anche in futuro di toccare con mano quelle che sono le preoccupazioni delle eminenze grigie della microinformatica, proponendosi, al contrario di altri saloni già esistenti, come un veritiero e attendibile centro di lavoro. E' questa una ambizione particolarmente sentita in materia di programmazione tant'è vero che si è instaurata una serie di conferenze studiate esclusivamente per i programmatori.

Tutto questo sta ad indicare che l'European Computer Trade Show intende dare di sé, già dalla sua prima edizione, un'immagine seria e imprenditoriale. Infatti esso è riservato ai professionisti della microinformatica ed esclusivamente ad



Come mostra il logo appositamente studiato per l'occasione, il nuovo salone ECTS nasce nel segno dell'Europa.

essi. Non sono qui ben visti nè gli amanti del shoot'em-up nè tutti coloro che operano nel mondo consumer dei games e del divertimento. Ma, così facendo, non si rischia un tantino di limitare il numero dei visitatori a pochi addetti ai lavori?

Commodore News

Al Sicob di quest'anno la Commodore ha presentato la sua nuova linea le cui punte sono l'Amiga 2500UX, la scheda 6830 e il package PAO.

Funzionante sotto sistema Unix V3.2, l'Amiga 2500UX è una potente postazione di lavoro multifunzione. Il sistema dispone di un disco rigido da 80 Mb con un tempo d'accesso di 19 ms e uno streamer da 150 Mb permette il salvataggio dei dati. I dati sono gestiti da un processore 68020 Motorola ad una frequenza di clock di 14,2 MHz. La gestione della memoria è affidata ad un processore supplementare, il 68851, mentre non poteva mancare il co-processore matematico 68881 a 25 MHz con la propria memoria da 4 Mb di RAM a 32 bit e 1 Mb di RAM a 16 bit. Ma Amiga 2500UX è ben più di una postazione di lavoro Unix, in quanto l'utente può accedere ai sistemi Amiga/DOS e Amiga/Unix attraverso il disco rigido

scegliendo l'ambiente entro il quale è necessario lavorare, selezionando semplicemente col mouse entro l'icona desiderata. Per mezzo di interfacce, il sistema offrirà il meglio della grafica e potrà colloquiare per mezzo di ben sette porte RS232 supplementari con Amiga 500-2000 e terminali dedicati PC, XT ecc. La scheda 68030 permette la creazione e l'animazione d'immagini sintetizzate. Destinata all'Amiga 2000, tale scheda assicura una gestione molto rapida dei dati avendo a disposizione un clock di 25 MHz. Le prestazioni grafiche di Amiga 2000 vengono incrementate del 600% il che pone il sistema sul mercato delle stazioni di lavoro grafiche. Il package PAO offre una soluzione professionale nell'ambito del Desktop Publishing. E' composto da un Amiga 2000 con 3 Mb, un disk drive da 3 e 1/2, da un hard disk con 40 Mb in autoboot, un monitor hi-rescon risoluzione 640x510 e la stampante laser NEC L890.

Morte a Venezia è un game combinato di adventure/azione prodotto dalla Cobra. Attendiamo con una certa curiosità anche la versione C64.

Morte a Venezia

Cobra Soft presenta una nuova Super-Produzione combinata di adventure/azione: Morte a Venezia.

Venezia è minacciata. L'ultimatum dei terroristi scade fra cinque ore! E' tempo che voi entriate in gioco per disattivare il meccanismo infernale.

Le schermate hanno colori splendidi (la città di Doges digitalizzata!) e troverete dozzine di personaggi, fateli parlare! Smascherate i colpevoli.... Analizzate gli indizi! Dovrete voi stesso usare dei trucchi! Commedia d'arte, alta tecnologia, trucchi, armi, veleni, mafiosi, cadaveri... sono alcuni degli ingredienti che compongono questa nuova Super-Produzione realizzata da Bertrand Brocard.

Non avete soltanto a disposizione un gioco appassionante, interamente grafico, ma tante attrezzature che vi permetteranno di avanzare: una vera "integrazione" che comprende il trattamento del testo, la gestione dell'archivio, il disegno logico, foto-album. E per preparare la fase finale del gioco, un neutralizzatore di bombe: un simulatore di montaggio elettronico!



Divagazioni

Stiamo per entrare in un'era nella quale il ruolo del docente, finora visto come esperto, figura autorevole e dispensatore del sapere, non sarà più lo stesso. Si vengono così a proporre nuovi modelli per questa figura: insegnante come guida, visionario (nel senso buono della parola), scopritore ecc. Gli insegnanti stessi stanno cercando di attuare cambiamenti che migliorino la qualità del loro ambiente di lavoro, che rendano più fruttuoso l'insegnamento agli studenti, e che elevino la loro posizione agli occhi della società e delle stesse istituzioni scolastiche.

Un programma di studi multimediale

Presto, sarà la stessa tecnologia ad offrire agli insegnanti l'opportunità di cambiare, ridefinendo il programma che verrà loro richiesto d'insegnare.

Entro il prossimo decennio, si trasformeranno anche i media della conoscenza, come la divulgazione, la radiodiffusione ed in genere tutti i campi della comunicazione che convergono ad un comune standard digitale. Radio e TV, cinema, telecomunicazioni, divulgazione affidata a libri, riviste e quotidiani, data processing e persino l'arte, tutti gli strumenti di comunicazione della epoca moderna (e pertanto anche coloro che ne sono i principali tenutari) convergeranno fino a sovrapporsi ed identificarsi. Ciò che ne risulterà sarà un'immagine "ibrida" dei rami del sapere, che si realizzeranno attraverso divulgazione multimediale, computing multimediale e così via. Allo stesso tempo si renderanno disponibili nuovi strumenti per questo tipo di comunicazione che diverranno mezzi indispensabili per la comunicazione interpersonale. Il significato del termine *Desktop Publishing* si estenderà fino ad includere la produzione di "documenti" che combinano immagini fotografiche, voci umane, musica ed effetti sonori, video e grafica computerizzata come pure parole e numeri.

Insegnamento e tecnologia negli anni '90. Cerchiamo di scoprire come si trasformerà l'immagine dell'insegnante nel prossimo decennio e a quali risultati questo cambiamento potrà portare.

Fred D'Ignazio

I nuovi dispositivi multimediali per la comunicazione ed il sapere eventualmente si restringeranno nelle dimensioni e nel costo fino a diventare adatti all'insegnamento scolastico.

Gli insegnanti che per primi abbracceranno queste tecnologie avranno l'opportunità unica di diventare pionieri ed esploratori di un nuovo e significativo ambiente per l'insegnamento nel quale gli studenti avranno la possibilità di ricostruire il corso di studi in formato multimediale. L'ambiente scolastico odierno, costituito esclusivamente dalla comunicazione orale e dal testo, si evolverà in una sorta di studio nel quale lo studente produrrà video, mostre di diapositive elettroniche, relazioni in video-libro, grafica d'informazione e giornali in termini multimediali. Le materie difficili ed aride diventeranno interessanti ed eccitanti quando lo studente le tradurrà in presentazioni multimediali.

Multimedia di recupero

La divulgazione multimediale per studenti diverrà un giorno normalissimo materiale di apprendimento e consultazione come lo è oggi un libro di testo. Comunque, gli insegnanti che vedono la possibilità di un insegnamento di questo tipo

autoprodotto dagli studenti, non devono aspettare un futuro molto lontano.

Possono cominciare ora. Possono portare l'ambiente di divulgazione multimediale degli anni '90 nelle aule di oggi, riunendo e recuperando i frammenti separati (costituiti dai diversi mezzi di comunicazione) in un singolo centro d'insegnamento, composto da attrezzature che si trovano già in molte scuole.

Gli insegnanti, quando vogliono, sanno essere improvvisatori ingegnosi, sanno recuperare materiali disparati, possono quindi sfruttare queste qualità per lavorare con i loro studenti e raccogliere tutti gli strumenti dei quali hanno bisogno: computer, videoregistratori, TV, telecamere, registratori a cassette, giradischi e tastiere elettroniche.

Gli studenti: "produttori" del proprio sapere

Gli studenti possono partecipare attivamente nel mettere assieme le diverse parti di tale sistema.

In un centro multimediale, i "giocattoli" elettronici come le telecamere, le tastiere musicali, gli altoparlanti ed i walkman, possono essere usati simultaneamente per dar luogo ad una stazione di divulgazione multimediale capace di produrre trasmissioni radiofoniche di scienza, biografie in video, progetti scientifici multimediali, e tutte le specie possibili di applicazioni eccitanti ed innovative.

Comunque, non accadrà niente di tutto ciò fino a quando i professori non intenderanno dividere le responsabilità dell'insegnamento e dell'apprendimento con i loro studenti. Questo può avvenire in un ambiente d'apprendimento dove ognuno impara dall'altro.

L'insegnante è uno specialista del proprio corso ed uno specialista del sapere in genere. I ragazzini sono particolarmente pratici e inseriti già di natura in queste nuove tecnologie e sono quindi capaci di

(segue a pagina 59)

Programmazione in l.m.

Molti programmatori BASIC evitano la noiosa istruzione INPUT scrivendo routine di input personalizzate che usano l'istruzione GET. Il problema è che, nel fare ciò, si incorre spesso in nuovi problemi, tipo i ritardi di garbage collection (collezione della "spazzatura").

Può un breve modulo in linguaggio macchina eseguire l'input senza gli svantaggi delle istruzioni BASIC? Sì, ma c'è una difficoltà. Un programma in LM può ottenere una stringa dal BASIC con facilità, ma spende moltissimo tempo nel passare la stringa al programma di richiamo BASIC. Le stringhe BASIC, sono complicate, e i diversi computer Commodore le costruiscono in maniera differente.

Quando un programma BASIC crea una stringa con un'istruzione tipo INPUT X\$, si verificano numerosi eventi complessi. Viene allocata memoria per X\$ nella tabella delle variabili BASIC. Il sistema immette la stringa in un'area buffer e ne misura la lunghezza. L'interprete BASIC cerca nella memoria uno spazio sufficiente per questa stringa. Se trova lo spazio, vi copia la stringa, e contraddistingue l'area come "usata".

C'è di più, vengono stabiliti gli indicatori. Se lo spazio nella memoria non si trova, avvengono più cose. Le stringhe BASIC possono apparire complicate dal punto di vista del programmatore LM (per non parlare dei procedimenti utilizzati dal 128 per evitare problemi di garbage).

Noi non vogliamo fare tutto questo nella nostra routine in linguaggio macchina, almeno non se si può evitare.

Una soluzione al problema è questa: lasciate che il BASIC faccia una stringa dovunque vuole nella memoria. Un programma in LM può ricevere l'input da tastiera e inviarlo direttamente alla stringa BASIC esistente. Quando il programma LM ritorna al BASIC, la stringa conterrà l'informazione.

Questo metodo viene spesso chiamato

Il passing delle stringhe

Jim Butterfield

"string thing". Lo useremo fra un momento, ma prima voglio specificare alcune cose alle quali faremo attenzione.

Dobbiamo essere sicuri di non oltrepassare l'area di immagazzinamento di una stringa. Se la stringa è stata generata dal BASIC con una lunghezza di 10 caratteri non dobbiamo accettare più di 10 caratteri.

D'altra parte, se la stringa d'input non è tanto lunga quanto lo spazio fornito dal BASIC, bisogna trovare un modo per comunicarlo al BASIC, dicendogli quanti caratteri abbiamo effettivamente inserito.

String thing

Osserviamo il codice di un semplice programma string thing. Questo programma non possiede un codice speciale per caratteristiche tipo riconoscere il tasto DELETE o fornire un segnale di sollecito.

Queste caratteristiche sono semplici da aggiungere.

Come è stato scritto, questo programma funziona solamente sul Commodore 64 o sul 128 nel 64 mode. Una versione per il 128 esigerebbe codice extra per consentire il banking della memoria.

La prima cosa da fare è trovare la stringa così da sapere quanto è lunga e dove è situata. Supponiamo che la variabile di stringa sia la prima variabile usata nel programma. Questo ci risparmierà del codice, visto che ci dice che i data della stringa saranno all'inizio della tabella variabili.

La tabella variabili inizia ad una locazione indicata negli indirizzi \$2D e \$2E (l'indicatore "start-of-variables"). La prima variabile è presupposta essere quella che vogliamo. I primi 2 bytes contengono il

nome della variabile. Noi non ne abbiamo bisogno, così inizieremo con un 2 nel registro y:

```
0380 LDY #$02
```

Questo loop copia i bytes nella tabella variabili negli indirizzi \$8B-\$8E. I primi 3 bytes sono di descrizione: \$8B dà la lunghezza della stringa, \$8C/D ne danno l'indirizzo, \$8E contiene uno 0.

```
0382 LDA ($2D),Y
      (elemento
della variabile)
      STA $0089,Y
      INY
      CPY #$06
      BNE $0382
```

Ecco il nostro principale ciclo GET. Noi prendiamo (GET) un carattere se è binario 0, ritorniamo immediatamente all'asserzione GET, visto che lì non è stato ricevuto nessun carattere. Poi controlliamo per vedere se il carattere è un RETURN, nel qual caso, lasciamo il loop:

```
038C JSR $FFE4 (GETIN)
      BEQ $038C
      CMP #$0D (RETURN?)
      BEQ $03A5
```

Abbiamo un carattere, e non è un RETURN. Otterremo il conto del carattere dall'indirizzo \$8E (inizialmente vi sarà uno 0), e controlleremo per verificare che non sia al limite della dimensione della stringa (il cui valore è depositato in \$8B). Se siamo al limite, ignoriamo il carattere:

```
LDY $8E (Conto)
CPY $8B (Limite?)
BCS $038C
```

Sembra essere un carattere accettabile. Lo

(segue a pagina 59)

SIM-HI-FI IVES



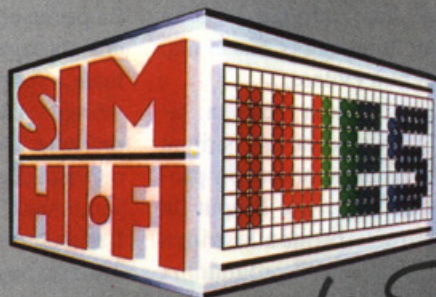
**23° salone internazionale della musica e high fidelity
international video and consumer electronics show**

14-18

settembre 1989

Fiera Milano

STRUMENTI MUSICALI,
ALTA FEDELTA', HOME VIDEO,
HI-FI CAR, CAR ALARM SYSTEMS,
PERSONAL COMPUTER, TV,
VIDEOREGISTRAZIONE,
ELETTRONICA DI CONSUMO.



un grande Sim!

Ingressi: Piazza Carlo Magno - Via Gattamelata - Orario: 9.00-18.00
Aperto al pubblico: 14•15•16•17 - Giornata professionale: lunedì 18 settembre



**VIVA
i giovani**

Segreteria Generale SIM-HI-FI-IVES: Via Domenichino, 11 - 20149 Milano - Tel.: 02-4815541 - Telex: 313627 - Fax 02-4980330



Scheda operativa

Pilota:	Capitano Ford
Condizione:	Sacrificabile
Vascello :	"PLASMATRON"
Missione:	Riconoscimento
Destinazione:	Loughton 2
Scopo:	Identificazione degli extra-terrestri

Considerazioni particolari

- 1) Pianeta distrutto e completamente frantumato
- 2) Presenza di extra-terrestri ostili
- 3) Punto di evacuazione della base ormai deserta.

"Ford non hai più nessuna chance" - sussurrava sorridendo felice l'alieno davanti al monitor.

"Buona fortuna!!!" gli augurava l'amico assistente
 "Grazie signore!" - aggiungeva Ford con un cenno di saluto, avviandosi verso Plasmatron..

La storia

Il motore ormai "gridava" impazzito per il caldo soffocante come un coro di angeli indemoniati.

"Hai un raggio vettore contro di te!" gridò Melker.

"Confermato!!"

Melker, l'assistente di Ford aggiunse: " Ustione 1, 0, 4 per cento, potenza al massimo"

Ford guardò fuori nello spazio: "Variazione eseguita !"

L'attimo successivo, si trovò circondato dalle stelle al sicuro dal pericolo imminente.

2 ore a 7 G... con questo passo avrebbe raggiunto la velocità di 10% fotoni (velocità leggera standard).

"A DRIVE +90" - specificò Melker

"Iniziamo l'operazione" - ordinò Ford.

La sua sedia incominciò improvvisamente a vibrare tanto che il suo corpo fu scosso da un lungo e profondo brivido. Persino i suoi abiti andavano sempre più irrigidendosi come il guscio di un insetto.

Ora egli era finalmente pronto per l'incredibile accelerazione che lo avrebbe portato al di là della velocità standard della luce. "+5 +4 +3 +2 +1"

Le galassie che esplodevano contro di lui sembravano birilli impazziti: solo allora si rese

conto del pericolo in cui si trovava.

"ARRIVO -1" - informò Melker

"Fisioreabilitazione, velocità; 10%, destinazione +300"

"Combattimento serrato.

Sistema offensivo attivato, livello d'alimentazione spray".

Ormai Ford poteva scorgere in lontananza il pianeta color grigio-blu.

Volò attraverso l'atmosfera del pianeta distrutto e sfiorando la superficie un tempo abitata, ma ormai in sfacelo.

"Stormo nemico con più di 5 tipi di astronavi, in avvicinamento a +10"

Flashback: La sua mente lo portò improvvisamente indietro

mostrandogli tutta la sua vita.

Come degli echi, facce amiche e care attraversarono

prepotentemente la sua mente azzerando totalmente i suoi pensieri.

"Stormo nemico in avvicinamento a +3 +2 +1"... il

secco annuncio lo riportò di colpo alla tremenda situazione

che stava per coinvolgerlo...

Piovevano navicelle aliene contro di lui come fosse grandine.

Ford sparava i suoi missili che

schiantandosi contro le navicelle aliene, le facevano esplodere producendo un bagliore biancastro accecante che illuminava l'interno della cabina rendendo l'ambiente ancor più allucinante di quanto già non fosse.

"Nuovo stormo nemico con più di 500 tipi di navette a +10"

"Accidenti" borbottò Ford.

"Niente da fare, stormo molto compatto a + 7,53"

Ford abbassò la cloche

lasciandosi cadere facendo sfiorare a Plasmatron il suolo irregolare della città ormai distrutta dagli alieni: un terribile incubo.

" +1 "

"Melker, ci rivedremo all'inferno!"

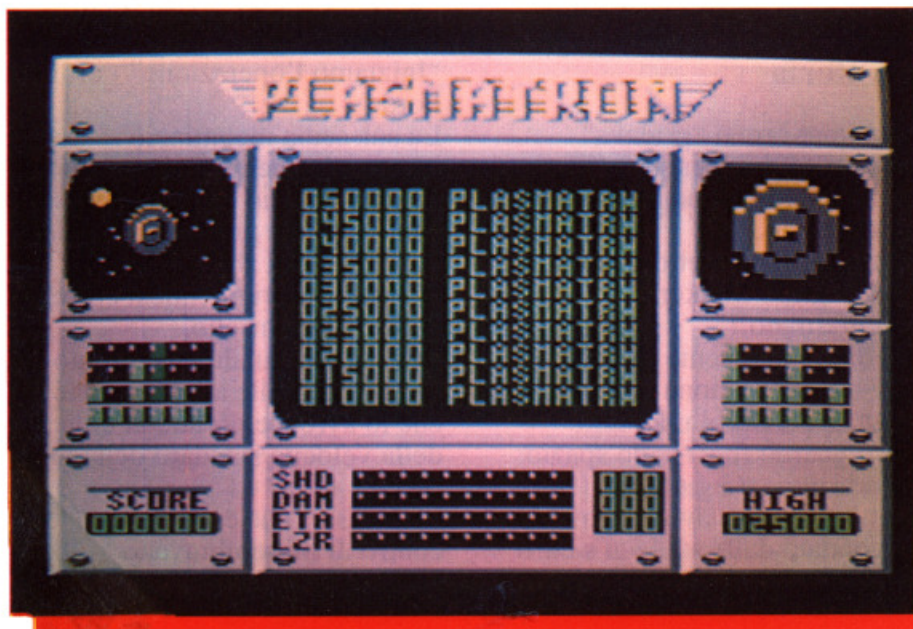
Il gioco

Al comando della nave Plasmatron, in effetti, non c'è il capitano Ford, bensì ci siete voi (!) che dovete controllare in ogni momento tutta la zona che state sorvolando, tenendo gli occhi ben aperti e i riflessi all'erta.

Lo schermo di apertura vi mostra sul monitor centrale la classifica dei punteggi raggiunti in precedenza da altri capitani che hanno tentato l'impresa prima di voi.

I due monitor laterali presentano invece le galassie in transito e l'avvicinarsi del pianeta, campo delle operazioni.

Lo schermo operativo presenta invece la vostra battaglia contro le navette spaziali nemiche che vi piombano addosso all'improvviso.





Gli scenari sullo sfondo sono i più svariati, potrete trovarvi a ridosso delle montagne, sopra città distrutte (attenzione a non volare troppo bassi) oppure nello spazio sconfinato: a seconda del luogo, varia anche la difficoltà.

Durante lo svolgimento delle operazioni, avrete sempre davanti a voi il cruscotto di Plasmatron posto nella parte bassa dello schermo.

Sulla sinistra troverete il punteggio attuale da voi riportato che terrà conto dei colpi andati a segno.

Dal lato opposto, un secondo segnalatore uguale a quello appena visto, vi terrà informati del punteggio record mai raggiunto: è questo l'obiettivo minimo che dovrete cercare di

raggiungere, se non altro per avere una, anche se magra, consolazione.

Al centro del cruscotto troverete quattro specchietti molto importanti per il proseguo della vostra missione: il primo, SHD, vi segnala lo stato degli schermi di protezione destinati a fermare, o quantomeno, ad attutire i colpi alieni, quando la barra raggiunge lo zero, rimanete senza difesa ed allora... problemi vostri...

Altro parametro è DAM, che vi segnala la percentuale di danneggiamento di Plasmatron, se non volete finire anzitempo... a piedi, cercate di non far scendere troppo questo livello.

ETA vi informa invece sul numero dei Plasmatron a vostra disposizione: partite con tre ship in dotazione.

Anche il laser LZR va, di tanto in tanto, ricaricato, per cui un suo uso eccessivo o sconsiderato, fa scendere il suo livello di energia, da mantenere sempre attivo per non rimanere disarmati.

Comandi

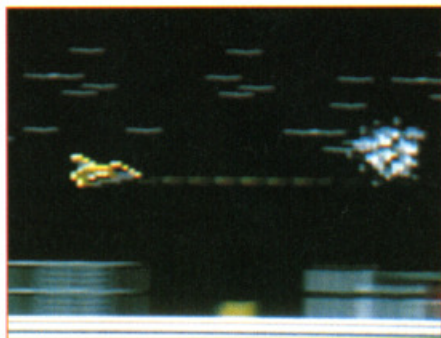
Per muovere la navicella spaziale, bisogna collegare il joystick in porta 2, naturalmente per far fuoco bisogna utilizzare il pulsante di fire. Se preferite utilizzare la tastiera, eccovi i tasti interessati:

Z	per spostarsi verso sinistra
X	per spostarsi verso destra
;	per spostarsi verso l'alto
,	per spostarsi verso il basso

Se volete concedervi una pausa, premete il tasto contrassegnato con la lettera F, per continuare la partita battete la lettera P.

Istruzioni di caricamento

Per caricare il supergame dal dischetto, digitate LOAD""",8,1 e premere il tasto RETURN, Plasmatron si caricherà automaticamente essendo il primo programma registrato nella directory. Per caricare il game da nastro, premere i tasti SHIFT-RUN/STOP e quindi mandare in PLAY il registratore, anche in questo caso il game partirà in maniera automatica.



ATTENZIONE:

prenotate in edicola il prossimo numero di Supercommodore !
Per voi il favoloso megagame

JET BOYS

Recensioni

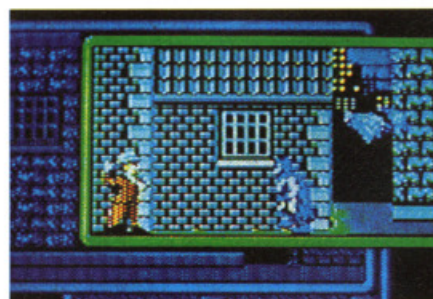
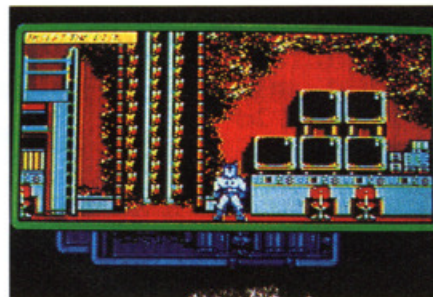
Batman

●	EDITORE:	OCEAN
●	GRAFICA:	9
●	SUONO:	8
●	AZIONE:	8

Siete molto pazienti? Amate le avventure difficoltose? Non siete dei tipi che si abbattono quando il castello di carte che state costruendo crolla miseramente nel momento in cui appoggiate sopra l'ultima carta? Allora ecco il vostro gioco, correte a comprarlo! Da questo game, devo dire, mi aspettavo qualcosa di avventuroso e difficile, un soft di non facile interpretazione, ma non fino a questo punto... Nel giro di dieci minuti di gioco, il povero Batman muore prima ancora di riuscire a percorrere il 6% della missione, molto promettente... insomma! abbastanza male. Decimo tentativo... decimo fallimento. Però, importante, questa volta, qualche progresso è stato fatto. Un caffè, uno stuzzichino, e voilà, pieno di coraggio, mi appresto al terzo tentativo. Guido Batman con calma da una parte all'al-

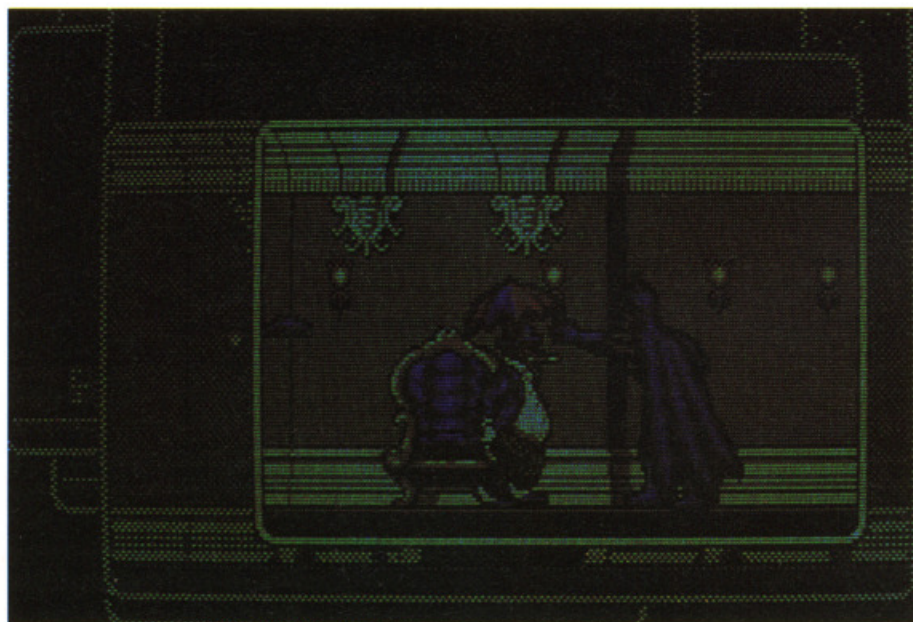
tra dello schermo e riesco a prendere gli oggetti che incontro (un'ampolla ed un boomerang), forse ci siamo: boom... Gli oggetti sono spariti e non hanno lasciato che un rebus: due vie chiuse e una aperta. Tutto il resto dello schermo è diventato nero! Non può essere stato che un errore di manovra... Tento allora di usare l'ampolla: ri boom... meglio controllare il soft, prima di scoraggiarmi, ma qui, niente da eccepire. Nella penombra della stanza, a tentoni, cerco allora qualcosa che assomigli a un interruttore e, finalmente, (meno male) ci sono: Batman preme l'interruttore, ce l'ha fatta? Ohooo!! 15% tutto insieme, fantastico! Però, ahimè niente luce, ma il congegno è sicuramente lì... Altro tentativo e finalmente... la torcia che accendo con il soffio illumina l'ambiente e mi porta subito al 32%.

Da questo momento il ritmo sale, tutta una serie di oggetti arriva senza sosta: due porte, una cellula, un pulsante, delle tenaglie, eccetera. L'obiettivo non è completo, ma con la luce posso esplorare l'ambiente in lungo e in lar-



go e sperare di farcela. Che dire ancora! Un pò di sorpresa non guasta... Posso dirvi che "Batman" metterà a dura prova il vostro senso di orientamento, in quanto le indicazioni sono estremamente rare.

Le schermate si assomigliano un pò tutte, questo però non significa che siano mediocri. Le scene sono piene di colore ed anche il suono non è da meno. La grafica è molto originale e il movimento scorrevole (manovrate con accortezza il vostro joystick o usate la tastiera) vi aiuteranno, con la vostra solita abilità, a sconfiggere i vostri più acerrimi nemici: Joker e Pinguin.



Total Eclipse

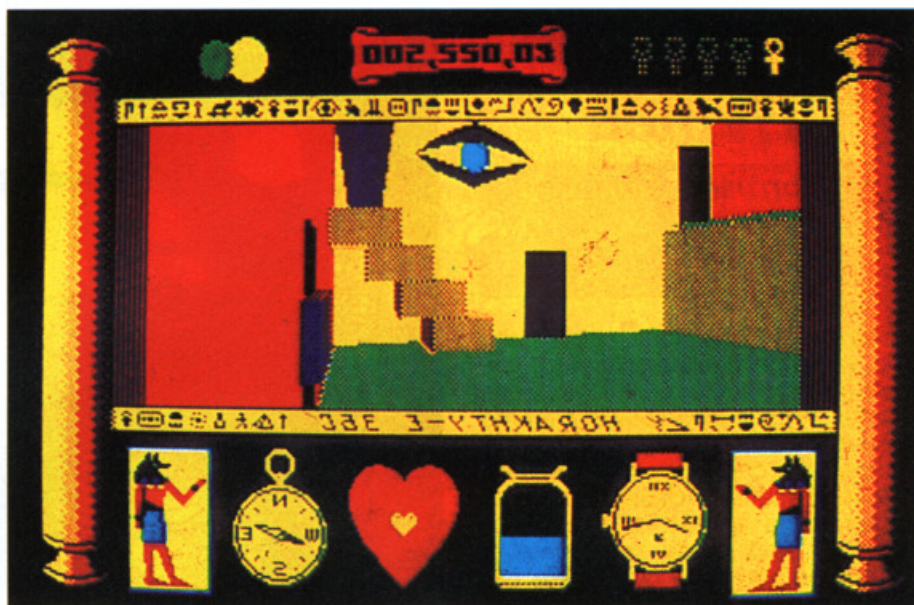
●	EDITORE:	INCENTIVE
●	GRAFICA:	8
●	SUONO:	8
●	AZIONE:	7

Che ne dite di un piccolo viaggio nel misterioso e affascinante Egitto? Il sole che scotta e la sabbia bionda a perdita d'occhio sono l'handicap, ma, per contro, una visita guidata alle meravigliose piramidi di Gizeh: siamo sicuri che accetterete senza esitazioni...

La malia dell'Egitto si è già impossessata di voi, infatti, la guida che sparisce improvvisamente imparendovi, non vi impedisce di entrare all'interno della piramide da soli. Qualcosa di più forte di voi, vi spinge all'interno della tomba in cui riposa uno dei più grandi faraoni della storia egiziana, in un labirinto di notevole complessità. La molla non è altro che il fascino della scoperta: questa piramide non è mai stata esplorata da nessuno prima di voi. In effetti, questo game, riesce a creare, grazie alla grafica adeguata ed alla musica un pò angosciante, una reale sensazione di suspense e di batticuore.

Voi sovrastate la scena e con l'aiuto della bussola, che non dovete perdere d'occhio, potete ricaricare la vostra energia (un liquido blu all'interno di una fiala) approfittando delle riserve che trovate in certe sale della piramide. Il vostro punteggio verrà, inoltre, incrementato ogni volta che troverete, al posto di figure con sembianze umane, delle pietre preziose. Sale, passerelle, passaggi segreti (che si apriranno allorchè utilizzate la vostra arma): ecco da qui in poi occupate le vostre lunghe serate, ma attenzione ad essere rapidi nel terminare il percorso prima che l'eclisse diventi totale (potete tenervi aggiornati sulle condizioni dell'eclisse guardando nello schermo in basso), oltre ai vari trabocchetti non sottovalutate mai la maledizione che ogni piramide che si rispetti ha in sé.

Total Eclipse è un labirinto realizzato brillantemente, con scene tanto vive da sembrare reali, infatti sebbene non eccella per fluidità di movimento, il fascino delle scene non mancherà di prendervi! Buona fortuna!

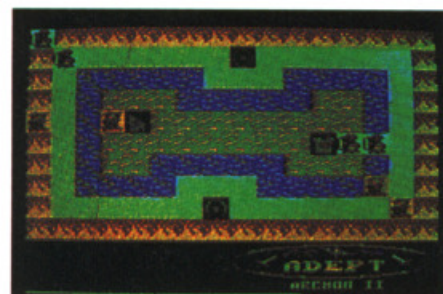


The Archon Collection

●	EDITORE:	ELEC. ARTS
●	GRAFICA:	8
●	SUONO:	7
●	AZIONE:	6

Electronic Arts ha cercato con The Archon Collection di dare una veste particolarmente originale a due nuovi games di strategia, introducendo nozioni e riserve di energia finora mai usati in questi particolari giochi (scacchi, dama). Lo scenario in cui si svolge il gioco vede in opposizione le tradizionali forze del Bene e del Male (perenne lotta): maghi, walchirie, eccetera. Nel menu due giochi: il primo, Archon, è costruito su una scacchiera con le solite caselle dove è possibile fare le proprie mosse, come

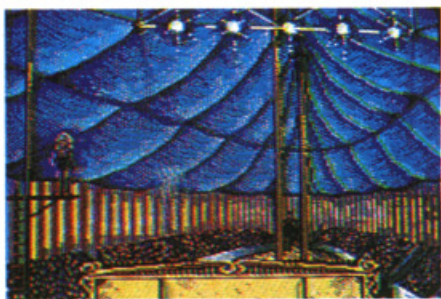
nel gioco degli scacchi. Qui, un rappresentante del Bene si ritrova nella stessa casella con una rappresentante del Male con cui deve ingaggiare una lotta all'ultimo sangue per sconfiggerlo. E' però consigliabile calcolare bene i rischi prima di dare duello al Male. La prima sfida è senza dubbio la più interessante per gli amatori di scacchi principianti, mentre nella seconda, che vede in lotta i quattro elementi naturali (fuoco, terra, aria ed acqua) sarà necessaria una maggior rapidità nei movimenti. Questo game, anche se gradito agli appassionati di giochi strategici, non è francamente geniale, anche se non manca di interesse.



Circus Games

●	EDITORE:	TYNESOFT
●	GRAFICA:	8
●	SUONO:	7
●	AZIONE:	8

Una scena molto piena, colori molto vari, valzer del buon vecchio Strauss, sono le particolarità di rilievo di Circus Games, vi conducono piacevolmente nella malia del mondo del circo. Le specialità che vengono rappresentate sono quattro: il classico trapezio, le acrobazie a cavallo, esercizi di equilibrio sulla corda senza rete di protezione con salti, giravolte e per finire, se non siete ancora morti, dovete dare dimostrazione di bravura con una piccola (si fa per dire) prova di coraggio: domare le tigri del Bengala. Ecco un gioco veramente originale per il nostro C64, realizzato ineccepibilmente, da giocare sia da soli che in compagnia.



Bio Challenge

●	EDITORE:	DELPHINE S.
●	GRAFICA:	9
●	SUONO:	8
●	AZIONE:	9

Ecco un game, della Delphine Software, veramente interessante e fuori dai soliti canoni. La grafica, l'animazione, il suono e gli scenari sono paragonabili alle Rolls del "capo" contro la 500 del dipendente. Dopo Space Harrier, un adattamento favolosamente geniale su computer, ecco Bio Challenge, che niente ha da invidiare al precedente, ve lo posso garantire. In



Bio Challenge voi siete un robot e vi aggirate in un mondo tormentato pieno di mostri inquietanti, di curiose piattaforme, di crepacci e... scopritelo da soli (se no dov'è il divertimento?)!! Lo scopo del gioco è il recupero di un amuleto diviso in quattro parti e per portarlo dovrete affrontare due tipi di mostri, non certo ben intenzionati. Per distruggere i primi mostri la difficoltà non è eccessiva: è sufficiente che vi avvolgiate su voi stessi ed essi sono spacciati. Ci sono però i cosiddetti mostri "rampanti", e qui il discorso cambia, infatti dovrete sudare per poterli sconfiggere e soprattutto non fare assorbire da questi la vostra energia. Un modo per annientarli c'è: quando siete sulla piattaforma, giocando d'astuzia, muovendovi in diverse direzioni per trarli in inganno, riuscirete a farli cadere sotto rendendoli innocui. Fate attenzione ai nemici che appaiono nel cielo, cercate di abbatterli otterrete così dei bonus. Avrete anche la visione del tempo supplementare prima della fine, un vantaggio in più che vi permette di valutare la situazione senza esporvi al pericolo, dandovi un'insieme della schermata utile ai fini del gioco. Sia a livello grafico che in generale questo game è impressionante: lo scrolling ha cinque livelli e possiamo annoverare ben 160 colori; ci sia concesso dire "non è male". Ma non è tutto, la qualità dell'animazione è senza dubbio il fiore all'occhiello, infatti la velocità è folle. A questo, direte voi, ci siamo abituati! Certo, ma per un cartone animato, per il nostro C64 è già un pò più difficile: il nostro robot ha i movimenti super-

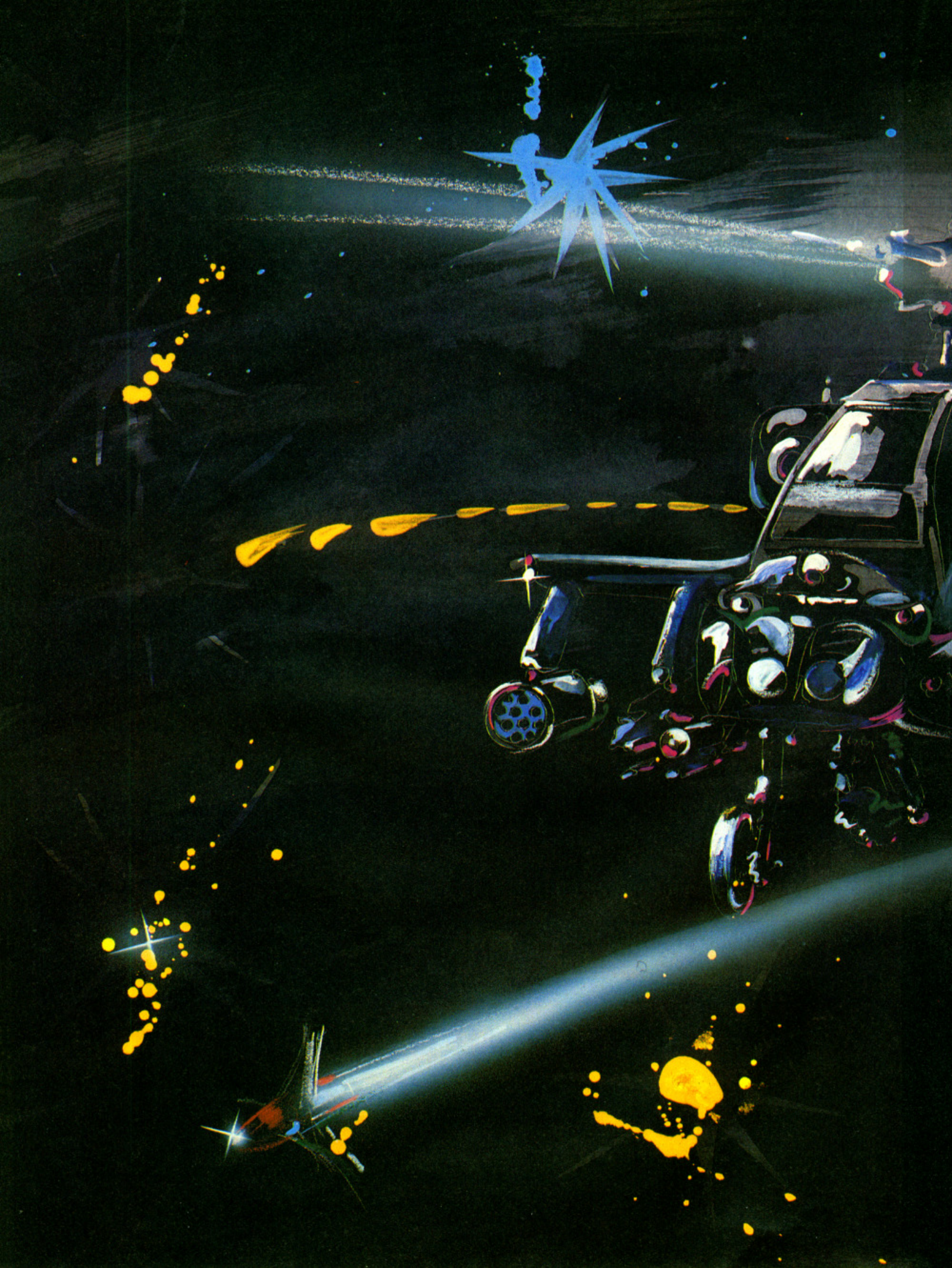
veloci. Salta e si muove come un essere umano, il che non è certo da robot. Superbo! Non ci sono altre parole per descriverlo. Anche il suono è all'altezza del game, veramente un buon arcade!



Superclassifica

a cura del servizio Soft Mail - Lago tel.:031/300174

1	DRAGON NINJA	<i>Ocean</i>
2	ROBOCOP	<i>Ocean</i>
3	TOTAL ECLIPSE	<i>Incentive</i>
4	ROCKET RANGER	<i>Cinemaware</i>
5	ITALY '90 SOCCER	<i>Italvideo</i>
6	G.PRIX CIRCUIT	<i>Accolade</i>
7	THUNDERBLADE	<i>U.S. Gold</i>
8	G.L. HOT SHOT	<i>Gremlin</i>
9	WHO FRAMED R.RABBIT	<i>Buena Vista Software</i>
10	MC ARTHUR'S WAR	<i>SSG</i>





GUIDA ALL'INPUT DEI PROGRAMMI

I programmi in BASIC listati su **SUPERCOM-MODORE** contengono una particolare codifica dei caratteri di controllo (cursore, colore, reverse, eccetera) che permette di ottenere una maggiore leggibilità dei programmi.

Generalmente, questi listati contengono alcuni caratteri racchiusi tra parentesi graffe {}; queste ultime, che non esistono sulla tastiera del computer, non devono essere digitate, ma hanno unicamente lo scopo di indicare che i caratteri da esse racchiusi sono dei caratteri di controllo.

Ad esempio {GIU'} indica che occorre premere una volta il tasto di cursore verso il basso, {3 GIU'} indica che il tasto di cursore verso il basso dovrà essere premuto tre volte.

Se tra le parentesi graffe è racchiuso un singolo carattere, quest'ultimo deve essere premuto insieme al tasto CTRL (ad esempio, incontrando {A} si

dovrà premere il tasto CTRL insieme al tasto A).

Invece i caratteri racchiusi tra parentesi quadre e simboli di maggiore e minore [< >] devono essere premuti unitamente al tasto Commodore (nell'angolo inferiore sinistro della tastiera).

Ad esempio, [<A>] indica che deve essere premuto il tasto Commodore insieme al tasto A.

Infine, alcuni caratteri racchiusi dalle parentesi graffe preceduti dalle lettere SH: ciò indica che il carattere seguente deve essere digitato tenendo premuto il tasto SHIFT.

La tabella seguente riporta, per ogni carattere di controllo, la codifica utilizzata nei nostri listati.

I listati in linguaggio macchina (totalmente numerici) non possono invece essere copiati direttamente con il computer, ma richiedono l'utilizzo dello speciale programma MLX, riportato, insieme alle istruzioni per il suo utilizzo, in altra parte della rivista.

TABELLA TASTI E CARATTERI

Quando leggete:	Premete:	Vedrete:	Quando leggete:	Premete:	Vedrete:	Quando leggete:	Premete:	Vedrete:	Quando leggete:	Premete:	Vedrete:
[CLR]	SHIFT CLR/HOME		[CYN]	CTRL 4		[C = 7]	CTRL 7		[CTRL G]	CTRL G	G
[HOME]	CLR/HOME		[PUR]	CTRL 5		[C = 8]	CTRL 8		[CTRL H]	CTRL H	H
[SU]	SHIFT < CRSR >		[GRN]	CTRL 6		[F 1]	F 1		[CTRL I]	CTRL I	I
[GIU']	< CRSR >		[BLU]	CTRL 7		[F 2]	SHIFT F 1		[CTRL J]	CTRL J	J
[SIN.]	SHIFT < CRSR >		[YEL]	CTRL 8		[F 3]	F 3		[CTRL K]	CTRL K	K
[DES.]	< CRSR >		[C = 1]	CTRL 1		[F 4]	SHIFT F 3		[CTRL L]	CTRL L	L
[RVS]	CTRL 9		[C = 2]	CTRL 2		[F 5]	F 5		[CTRL M]	CTRL M	M
[OFF]	CTRL 0		[C = 3]	CTRL 3		[F 6]	SHIFT F 5		[CTRL N]	CTRL N	N
[BLK]	CTRL 1		[C = 4]	CTRL 4		[F 7]	F 7				
[WHT]	CTRL 2		[C = 5]	CTRL 5		[F 8]	SHIFT F 7				
[RED]	CTRL 3		[C = 6]	CTRL 6		[FRS]	←				

SUPER COMMODORE 64/128

● Alien Syndrome

● EDITORE:	ACE
● GRAFICA:	8
● SUONO:	8
● AZIONE:	8

Uno dei soft più difficili con cui misurarsi attualmente. Per salvare i vostri compagni prigionieri degli alieni, voi mandate il nostro eroe all'interno di un elaborato labirinto. Per un istante sembra tutto tranquillo, fin troppo! Non ci metterete molto ad accorgervi che le apparenze, come ben si sa, delle volte possono ingannare! Ahimè! Alien mai visti prima ed in numero astronomico, sparano sul nostro eroe bombe radiocomandate.... Cercate di precederli e sparate senza sosta in tutte le direzioni contro questi mostruosi nemici. Quando siete vicini a un prigioniero, non indugiate, liberatelo subito da questo inferno. Se avete cara la pelle, non potete dormire sugli allori, i mostriciattoli non stanno lì a guardare, ma cercheranno di prendervi alla sprovvista. In vostro aiuto, (collegandovi a uno dei pannelli che trovate in ogni sezione che attraversate) premendo un pulsante del pannello, avrete armi più potenti e sofisticate oppure dei punti di bonus. Alien Syndrome, mette a dura prova la vostra capacità visiva nel rilevare la posizione degli alieni prima che abbiamo la possibilità di lanciaarvi contro le loro micidiali bombe, è di importanza vitale precederli. Da non sottovalutare la presenza del mostro che vi attende alla fine di ogni livello. Alien Syndrome è stimolante, la difficoltà è paragonabile al suo parossismo. Riuscire a portare in salvo tutti i prigionieri richiederà molto impegno e molti tentativi tanto da essere un po' frustrante. La grafica è accettabile così come lo sono l'azione e la velocità di gioco.



● Hawkeye

● EDITORE:	THALAMUS
● GRAFICA:	9
● SUONO:	8
● AZIONE:	8

Scena: il nostro povero pianeta è stato invaso da un esercito alieno. Gli abitanti terrorizzati si sono nascosti nelle viscere della terra, dove hanno messo a punto una creatura, mezzo uomo, mezzo robot, l'unica in grado di opporre una buona resistenza sui nuovi implacabili nemici che, tanto per cambiare, vogliono conquistare il pianeta. E' venuto il momento di passare all'azione! La creatura o il robot? È pilotata da un capitano con i nervi d'acciaio, voi!, buona fortuna! Lo schermo mette in risalto il quadro comandi del vostro vascello, la scena esterna, con colori ombrosi ma adeguati, rendono molto bene, a voi che giocate, la sensazione di un ambiente angosciante e oppressivo. Hawkeye è un match veramente difficile. Più si va avanti nel gioco, e più aumenta la difficoltà, senza tregua dal primo all'ultimo istante. Farete bene a sparare in tutte le direzioni senza mai smettere, se volete avere la certezza della vittoria finale! Potete però collezionare dei bonus speciali, che vi daranno una mano nella lotta; questi bonus possono sortire due effetti: distruggere i vascelli (esplodendo distruggono tutti i nemici presenti nella schermata), oppure aumentare il vostro arsenale dandovi in dotazione armi più potenti. Hawkeye è un game di azione pura e quindi mette a dura prova i vostri riflessi. La grafica e l'animazione non sono il massimo; mentre i colori molto vari e la velocità di gioco, saranno sicuramente molto apprezzati! In bocca al lupo....



● Druid II (Enlightment)

● EDITORE:	FIREBIRD
● GRAFICA:	9
● SUONO:	8
● AZIONE:	8

La prima versione di Druid aveva già sedotto un buon numero di avventurieri: era inevitabile una seconda versione adeguatamente rivista e migliorata. Ed ecco Druid II che, conosciuto anche come Enlightenment, rilancia la palla con un surplus di strategia apprezzabile. Gli appassionati di questo genere di giochi si renderanno conto subito di avere a che fare con uno dei migliori soft adventure/azione disponibili oggi sul mercato. Prima di tutto, la caratteristica principale del gioco, ovvero la fluidità delle azioni: i personaggi, piccoli druidi assai potenti, corrono a cento all'ora (si fa per dire!) tra imboscate, mostri, trappole magiche.... E' vostra la responsabilità di uscire indenni da questo inferno; per farlo dovete ricorrere all'azione rapida del joystick: se non lo azionerete con la dovuta rapidità, non avrete alcuna possibilità di scampo (requiem!). Un aiuto comunque lo avrete in quanto all'interno delle tasche del vostro mantello (in basso sullo schermo) potete trovare dei preziosi talismani, oggetti magici, che usati opportunamente, vi daranno: invisibilità, armature, eccetera; sono più o meno rari ma, senza dubbio molto utili, perciò vanno amministrati con parsimonia e in circostanze ben specifiche. Potrete anche avere dei nuovi compagni che creeranno voi stessi, essi vi aiuteranno (proteggendovi) in certi frangenti a salvarvi la vita. Druid II è senza dubbio un game di buona qualità sia per la fluidità delle azioni che per gli effetti grafici degli scenari!



● Operation Wolf

●	EDITORE:	OCEAN
●	GRAFICA:	9
●	SUONO:	8
●	AZIONE:	8

Operation Wolf fa un ingresso trionfale nel mondo dei micro, dopo essere stato il numero uno dei giochi arcade. In un paese dell'America Latina, colpito dai tormenti che genera abitualmente un colpo di stato, sono state rapite alcune personalità diplomatiche. Si sa che esse vengono tenute in ostaggio in un campo di concentramento situato nel cuore della foresta amazzonica. Un attacco in forza non farebbe che mettere a repentaglio la vita degli ostaggi. Invece l'infiltrazione di un commando è giudicata più adatta alla situazione. L'eroe siete voi, naturalmente, ed avete l'obiettivo di localizzare il campo di concentramento, liberare gli ostaggi e condurli all'aeroporto per prendere l'aereo e trasportarli al sicuro. Bazooka in mano, un pò di munizioni, qualche bomba, e, in cammino per l'avventura! Il gioco comporta sei livelli che si collegano logicamente. In un primo tempo, siete paracadutati in piena giungla con l'obiettivo di distruggere la rete di comunicazione, affinché il nemico non possa dare l'allarme. In caso di successo, dovrete esplorare la giungla per catturare i ribelli e ottenere le informazioni sul campo dei prigionieri.

Lungo la strada, il vostro grande cuore di cavaliere non resiste: capitate in un villaggio occupato dai ribelli e vi premurate di liberarlo. Sensibile all'accoglienza calorosa ed alla riconoscenza degli abitanti, decidete di fare una sosta. Di nuovo fresco e in forma siete pronto per affrontare il deposito di munizioni e rifornirvi. La tappa successiva vi porta al campo di concentramento dove sono tenuti prigionieri gli ostaggi. Il gran finale si svolge allo

aeroporto, dove l'aereo trasportatore vi attende con i vostri ostaggi liberati. La parte destra dello schermo è riservata ai dati che vi informano in continuazione sullo stato della vostra salute, il numero delle munizioni, dei nemici da abbattere, eccetera. Il resto dello schermo (i tre/quarti) costituisce la parte principale, offrendo una visione in primo piano. I ribelli corrono, si arrampicano, si gettano col paracadute, si scatenano per farvi la pelle!

dovete tenere sotto controllo tutti gli strumenti che concernono il volo automatico, dall'altro, manovrando con il joystick, le fasi attive: decollo, esplorazione del suolo lunare, eccetera, sono nelle vostre mani! Le vostre capacità di calcolo alla frazione di secondo, faranno di voi un eroico astronauta, ne siamo certi! Appolo 18 è senza dubbio il meglio che possiate trovare come simulatore di volo per il vostro C 64, che cosa aspettate....?

● Appolo 18

●	EDITORE:	ACCOLADE
●	GRAFICA:	8
●	SUONO:	7
●	AZIONE:	8

Torniamo al 21 giugno 1969, lo slanciato Appolo sventa altero nel cielo, tra le braccia di acciaio che lo sostengono, a Cape Canaveral. Voi siete già a bordo, pronto ad decollo, per iniziare quella che è (senza dubbio) la più grande avventura del secolo! La messa a punto di questa straordinaria simulazione per il vostro Commodore 64 è estremamente audace ed il risultato (garantito) non deluderà le vostre aspettative. Le azioni scorrono sistematiche e precise e i vostri occhi seguono attenti tutte le varie fasi: preparazione del veicolo, preparazione al decollo, rispetto della traiet-



toria, allunaggio ed infine, dulcis in fundo, esplorazione del suolo lunare. Le schermate si possono definire sontuose, cosa dire: C64 delle meraviglie! La grafica e la qualità dell'azione è talmente bella da sembrare reale. Il suono non è da meno, stessa qualità: il rimbombare dei motori al momento del decollo è una piccola meraviglia! Coraggio ora tocca a voi! Date prova di fare onore a tale capolavoro con la vostra strategia, portate a termine l'impresa curando ogni manovra con nervi saldi. Da un lato,

● Gauntlet II

●	EDITORE:	U.S. GOLD
●	GRAFICA:	7
●	SUONO:	9
●	AZIONE:	8



Gauntlet II, tutti lo riconoscono, è diventato geniale! Ma quello che lo distingue dalla versione precedente è la maggiore difficoltà: veramente notevole (quale sfida! niente appassiona di più i nostri smanettoni).

Una delle innovazioni di maggior spicco è la possibilità di gioco per più persone, mentre per quello che riguarda il suono, è l'inserimento di una serie di voci digitalizzate che accentuano l'ambientazione del gioco rendendolo più reale. I commentatori vi guideranno nella ricerca degli oggetti nascosti oppure (speriamo che non succeda) in caso di fallimento, annunceranno l'imminenza della vostra morte. Sia lo scrolling che la realizzazione grafica sono... sensazionali!!! Grazie ai vostri riflessi, siamo sicuri, riuscirete a passare indenni all'attacco dei mostri che vi si lanciano contro come kamikaze. La strategia non si discosta molto dalla precedente versione di Gauntlet, però vi troverete davanti una nuova serie di sortilegi e nuovi mostri: volatili, draghi, spari che rimbalzano, eccetera. Per il nostro C 64 un game di azione pura, una vera delizia!





FUTUREWARE

ovvero

**fantascienza su disco.
Il software di fantascienza
aumenta sempre più.
Alcuni nuovi prodotti,
ed altri classici
ormai famosi, mostrano
come alcuni programmatori
siano ormai prossimi
a riuscire a creare
dei veri mondi
fantascientifici su disco.**

Keith Ferrel

Spazio-novelas

Le spazio-novela costituiscono la forma di fantascienza più familiare alle masse e quella che è stata resa più popolare. Gli elementi base delle spazio-novela sono semplici: civiltà lanciate lontanissimo, mitiche navi spaziali e disperati conflitti per il destino dell'universo. Questi elementi possono essere combinati in modi

completamente diversi: Star Trek è una spazio-novela, ma lo è anche Dune.

La prima grande spazio-novela è stata Elite della Firebird. Con il suo vastissimo ambiente galattico, un contesto politico e commerciale realizzato sorprendentemente bene, combattimenti che andavano oltre il tradizionale gioco arcade. Elite ebbe un notevole seguito. Continua a divertire i fans e a stimolare imitazioni. Debitore verso Elite, anche se meno ambizioso nel suo progetto, è Star Empire della First Row.

Qui dovete attaccare stazioni spaziali, esplorare pianeti e cercare di costruirvi un impero. Più simile all'arcade rispetto al miglior software, Star Empire nondimeno possiede un certo fascino e vale la pena di essere provato.

Elite è stato anche uno dei primi giochi a fare molto affidamento sulla documentazione per rendere il proprio contesto fantascientifico più concreto.

Tale contesto, l'ambiente fisico, lo sfondo

Stelle lontanissime e frontiere remote, il limite dell'infinito ed il giorno che verrà dopo il domani. Di questi piaceri giovano i fruitori della fantascienza. Molti di questi piaceri sono ora disponibili su disco, pronti a trasformare il vostro C64 o C128 in una navicella spaziale, una macchina del tempo o una porta per un'altra dimensione.

E il meglio del software fantascientifico, come faceva il meglio della letteratura sua antenata, tratta la sua materia, ed il suo pubblico, seriamente. Nei modi e negli approcci, il software di fantascienza è vario quasi quanto lo è la stampa. Vi sono giochi e simulazioni per tutti i gusti, dagli adattamenti di racconti best-seller a ricostruzioni della navetta Enterprise, lunghi giochi di conquiste e sfruttamenti galattici, e combattimenti più veloci della luce sui campi di battaglia di domani, o selvagge visioni di pianeti sottosviluppati. In breve, il software di fantascienza è un prodotto il cui momento è arrivato.

politico ed economico, è per molti versi l'essenza della fantascienza.

Tuttavia le abbreviazioni convenzionali della fantascienza stampata, funzionano meno bene nei giochi per computer. Per stabilire queste convenzioni, i produttori usano confezioni, manuali, libri di suggerimento ed altri materiali esterni per dare un senso di veridicità ai loro prodotti.

Elite, per esempio, includeva non soltanto un manuale per il commercio, ma anche una adeguata (ma non spettacolare) novella di Robert Holdstock, un affermato scrittore di fantascienza.

Prodotti confezionati

Questa tendenza alla confezione che crea il contesto è giunta agli estremi nell'ultima generazione dei giochi tipo spara e fuggi. Molti di questi giochi si presentano oggi come simulazioni di missioni disperate nelle quali un pilota deve liberare l'universo da enormi vascelli alieni. In realtà, si rivelano spesso poco più che variazioni della terza generazione di giochi come Asteroids, Breakout e Space Invaders, niente a che vedere con la fantascienza. Questo non significa che non ci siano legittime simulazioni di veicoli fantascientifici. Esiste oggi tutta una categoria di software che, prendendo spunto da riviste e libri di fantascienza, è orientato verso l'hardware. Questi programmi vi mettono ai controlli di macchine del tem-

po, che vi spediscono nel bel mezzo di grandi battaglie da incubo. Anche se non sono veri e propri giochi fantascientifici (sono più fantasiosi simulatori di volo) i migliori di questi prodotti tendono ad avere almeno una certa coerenza.

Starglider, della Firebird, come Elite, è accompagnato da molto materiale stampato, tutto volto a stabilire un contesto credibile dal quale possano attingere le super-armi del gioco.

Deep Space della Sir-Tech fa buon uso anche della confezione: il prodotto è riposto dentro una specie di contenitore di sicurezza che contiene anche fac-simili di comunicati ufficiali futuristici del governo (Deep Space guadagna anche dei punti perché aderisce alle leggi fisiche più del normale).

Echelon, della Access, ha aggiunto un elemento sonoro alla guida dell'astronave, il Lip-Stik, un microfono connesso ad una cuffia che riceve ordini vocali. Il Lip-Stik funziona quasi *troppo* bene (un sospiro di sollievo dopo uno scontro terminato positivamente può anche dare il via ad un'altra raffica di missili) ma è la combinazione Echelon/Lip-Stick che diventa, insieme a Deep Space, una delle migliori simulazioni spaziali degli ultimi anni.

Starfleet I della Interstel dimostra che non è necessaria l'animazione grafica per rendere superba una simulazione di volo interstellare. L'interesse in questo caso è

nei sistemi, che offrono più di una dozzina di possibilità al vostro comando. Lo scenario del gioco è ben progettato, e vi vede nel bel mezzo di una disperata battaglia (come poteva essere altrimenti?) contro i Krellers e gli Zaldrans, due specie di alieni implacabili, ognuna con le proprie strategie di guerra. Forse proprio perché il gioco pone l'accento più sui sistemi e sulla strategia che sulle immagini, consideriamo Starfleet I uno dei più soddisfacenti simulatori di guerre spaziali.

Finalmente... un clone!!

Ma in tutti i prodotti di fantascienza, anche quelli maggiormente orientati verso l'azione, verso il gioco d'arcade, c'è sempre qualcosa di più che la semplice battaglia.

Con una trama meravigliosamente fosca, degna di una novella di Edmond Hamilton o Jack Williamson, Captain Blood della Infogrames, prende in esame uno degli argomenti più significativi e misteriosi della fantascienza: la comunicazione con culture extraterrestri.

Captain Blood narra la storia di una razza in lotta con il tempo, i giocatori sono in gara con un orologio che avanza inesorabile, mentre devono ritrovare cinque cloni rapiti. I cloni riproducono proprio i giocatori, e man mano che il tempo passa, prosciugano loro le forze vitali; trovati o sarete vittime di una degradazione biologica. Questo possiede veramente un'interfaccia fantascientifica. Voi vedete il gioco con gli occhi di Captain Blood. Lo schermo è riempito dai controlli della nave. Il joystick controlla i movimenti della mano di Captain Blood, basta un dito per attivare qualsiasi sistema. Questi sistemi sono potentissimi: possono inviare dei ricognitori, analizzare mondi, distruggere interi pianeti. La nave è pronta a portarvi, attraverso un coloratissimo iperspazio, a spasso per galassie piene zeppe di pianeti. Ma quali mondi danno rifugio ai cloni scomparsi, e quali invece non sono altro che rocce senza vita?

Per scoprirlo dovete iniziare un dialogo con gli alieni. All'inizio del gioco vi trovate in orbita intorno ad un pianeta sul quale dovete inviare un ricognitore.

Dopo aver teleguidato il ricognitore attra-





verso una lunga e ventosa vallata (un aspetto arcade che sfortunatamente deve essere ripetuto su ogni pianeta che si visita) raggiungete un luogo d'atterraggio dove, nei pianeti abitati, vi attende un alieno. Usando un linguaggio ad icone, inizia la comunicazione. A seconda della specie alla quale appartiene, l'alieno può assumere un atteggiamento favorevole, ostile o incrompensibile. Le vostre reazioni devono spingere l'alieno a rivelare le informazioni che cercate.

Non è facile, non più di quanto sia facile farsi capire in una regione straniera. Comunque, con più di 100 "parole" a vostra disposizione ed un pò di pratica, il vocabolario ad icone diventa uno strumento abbastanza flessibile che vi aiuta a risolvere il mistero del gioco e vi insegna anche qualcosa sulla natura della comunicazione. Captain Blood è fantascienza pura: visioni grandiose, colori brillanti, trama intrigante, e qualcosa da dire.

Raggiungere le stelle

Nessun prodotto coglie l'immensità dell'esplorazione e della conquista galattica come Reach for the Stars della SSG. Questo gioco pretenzioso ma soddisfacente vi sfida a dirigere l'espansione di una razza da un singolo pianeta a decine di mondi. Ora nella sua terza versione, Reach for the Stars è stato il primo gioco della SSG, e

rimane sicuramente il più ambizioso.

Un paio di fattori rendono Reach for the Stars veramente brillante. Una è il suo senso del bilancio. I giocatori manipolano economie e colture planetarie per mantenere la produzione a livelli soddisfacenti. Il gioco ha anche il senso della dimensione.

Il suo spazio è immenso, e rende la sensazione delle proporzioni galattiche magnificamente. Dove altri giochi sparpagliano sistemi stellari, o intere galassie, senza tanto riguardo per la realtà astrofisica, Reach for the Stars offre un universo credibile con pianeti credibili che rientrano in una varietà di categorie. Ai livelli più avanzati l'aderenza alla realtà aumenta: le stelle possono diventare nova, disastri naturali ed epidemie distruggono le civiltà, le distanze interstellari diventano ancora più difficili da attraversare.

Infine c'è l'interfaccia.

I menu di Reach for the Stars rendono facile inoltrare combinazioni complesse di comandi, lasciandovi liberi di contemplare la grandezza della vostra impresa galattica. Reach for the Stars non è per il giocatore occasionale, è un gioco che si avvicina parecchio alle innumerevoli possibilità e sfide con le quali Isaac Asimov, Paul Anderson ed altri autori delle storie sull'impero galattico hanno lottato per anni.

Continuando coraggiosamente

La più vecchia e celebre spazio-novela è sicuramente Star Trek. Kirk, Spock, McCoy e tutto il resto dell'equipaggio, da quando hanno debuttato in televisione hanno avuto un seguito interminabile: li possiamo tutt'oggi ammirare nei nuovi film, in quelli che ancora vengono proiettati nelle sale di II visione, nei libri e, ovviamente, su disco.

"Star Trek : The Kobayashi Alternative" è un'avventura interattiva che presenta un Enterprise realizzato stupendamente, insieme al suo equipaggio e all'universo attraverso il quale viaggia. Voi interpretate James T. Kirk, e impartite gli ordini agli altri membri i quali, grazie allo smart parser e ad uno schermo a finestre, sono molto simili alle proprie controparti cinematografiche.

L'universo di Kobayashi è abbondantemente popolato da pianeti ed alieni. Come tutte le avventure a testo, anche questa nasconde un segreto nel profondo, ed è un buon segreto. Il rapporto con gli alieni è ben realizzato, come del resto i principali sistemi dell'Enterprise.

Meno efficace è stata la seconda avventura di Simon & Schuster ispirata al telefilm: The Promethean Prophecy, che mi è sembrata un'avventura abbastanza semplice: tutti giochi di parole e poco a che vedere con Star Trek. Ma i fans di Kirk e compagni potranno tra poco divertirsi con Rebel Universe, che sarà già in circolazione quando leggerete questo articolo (inoltre non saremo sorpresi se il recente e crescente successo del film "Star Trek: The Next Generation" venisse tra non molto trasportato su disco).

Star Trek ovviamente non scomparirà.

Marte

Non tutte le vicende fantascientifiche sono ambientate così lontano da casa. Se c'è un pianeta che può essere definito la quinta essenza della fantascienza è proprio Marte, che sta a "due passi" da noi. E' stato lo scenario di decine, se non centinaia di storie, libri e film, ed infine è approdato anche sul C64 con Mars Saga, dell'Electronic Arts.

Non si tratta delle novelle marziane di Barsoom, scritte da Edgar Rice Burroughs



o delle Chronicles di Roy Bradbury (che comunque potrebbero ispirare giochi meravigliosi). Mars Saga presenta invece un mondo oppressivo ed aspro, che ricorda per certi versi il mondo di frontiera che Robert A. Heinlein ha illustrato in varie novelle. Oltre alle difficoltà di questo nuovo mondo (Marte è ostile ed inospitale) bisogna affrontare l'elemento umano. I giocatori di Mars Saga, un role-playing game, devono riunire una squadra di avventurieri e poi guidarli attraverso pericoli naturali e artificiali. Si tratta di un gioco realistico sulla colonizzazione del sistema solare. Mars Saga dimostra anche quanto efficacemente la tecnica del role-playing, standard nelle avventure di fantasia, possa essere usata per creare avvincenti esperienze di fantascienza.

Il mondo dei libri

Gli scrittori di fantascienza avevano previsto l'era del computer. Detto ciò, è un po' sorprendente che questi scrittori non siano stati coinvolti per tradurre i loro lavori o creare nuove visioni su disco. Qualche anno fa, infatti la attività si fece frenetica, con giochi basati su The Robots of Dawn, Fahrenheit 451 di Bradbury o su Rendezvous with Rama di C. Clarke che apparvero tutti in un breve lasso di tempo. Sfortunatamente questi giochi non infiammarono il mondo. Ultimamente invece, un paio

di prodotti sembrano voler dire che gli autori di fantascienza hanno ritrovato la giusta via. Parecchi mesi orsono, la Electronic Arts fece uscire la versione di Amnesia di Thomas M. Disch per il C64. Pur non essendo un prodotto di pura fantascienza, Amnesia riflette una certa sensibilità in questo senso. E non è sbagliato dire questo, perché Disch stesso è tra l'altro un eccellente e controverso scrittore di fantascienza. La grande novità, comunque, è stata la realizzazione, da parte della Interplay, di Neuromancer, un gioco basato sull'omonimo libro di William Gibson. Poche sono state le novelle che hanno fatto notizia come Neuromancer. Ambientato in un torvo quartiere di Tokio in un futuro molto prossimo, il libro ritrae un mondo ombra di computer hackers (i pirati del computer) chiamati cyberspace cowboys, un mondo pieno di dominazione corporativa, avidità, doppi giochi e così via. Così efficace, per non dire assolutamente originale, era Gibson nel manipolare questo materiale, che divenne il decano di un'intera scuola di scrittori di fantascienza, conosciuti come cyberpunks. La Interplay ha trattato il materiale di Gibson seriamente, realizzando una visione credibile, e per la gran parte coerente, di un futuro non di certo auspicabile. Facendo ciò, la Interplay ha dato un notevole contributo a gettare le

fondamenta per il futuro del software fantascientifico.

Il futuro del futuro

Con l'arrivo di Neuromancer, infatti, il software di fantascienza ha fatto un grande passo avanti verso la maturità. Ha ancora tanta strada da fare, ma alcune direzioni sono già state stabilite. Come dimostra Neuromancer, è possibile racchiudere moltissimo in un paio di dischi. Nei prodotti dei prossimi anni, l'ambiente fantascientifico si farà più ricco, e questa ricchezza si manifesterà in un paio di modi. Primo, vedremo più contesto su disco, nel gioco stesso. Usando database, bulletin board, ed altre tecnologie d'informazione come sfondo, Neuromancer stabilisce la natura del suo mondo mentre si gioca e non mentre si legge il manuale. Secondo, vedremo un maggior rigore in alcuni giochi. Reach For The Stars mostra che un gioco può prendere abbastanza seriamente l'universo e le sue regole (con qualche eccezione, come i viaggi più veloci della luce) e costituire ugualmente una divertente e piacevole esperienza. Gran parte dell'attrazione che esercita la fantascienza, è dovuta alla possibilità di manipolare distanze astronomiche e periodi di tempo geologici. Stiamo già vedendo una maggiore varietà di software fantascientifico. Oltre ai prodotti qui menzionati, negli ultimi mesi sono nati: il mondo post-olocausto ben rappresentato da Wasteland (Electronic Arts), un campo di battaglia del XXI secolo in Fire-zone (PSS), un futuristico mestiere di detective in Techno Cop della U.S. Gold. Ovviamente c'è ancora spazio per altri generi di divertimento oltre a quello dei giochi di conquista. Quello della fantascienza è un grande campo dove, con molta ricerca e duro lavoro, quasi ogni speculazione può essere resa credibile: vedremo speculazioni più grandi negli anni a venire. La fonte migliore alla quale attingere è l'immenso corpo della fantascienza stampata. Dozzine di novelle e di storie già esistenti farebbero da scenari perfetti per piacevoli programmi. Ci vogliono soltanto uno o due fabbricanti avventurosi, ed alcuni progettatori e programmatori che capiscano le esigenze particolari della fantascienza. Il pubblico è già qui che aspetta...



The Great Arcade Machine

Hubert Cross

Vi è mai capitato di avere un'idea brillante per un gioco arcade? Se avete incontrato dei problemi nel realizzare la vostra idea con il computer, The Great Arcade Machine fa giusto per voi. Usatelo per creare giochi come Pinball, PacMan, Space Invaders, o un'infinita varietà di tanti altri: l'unico limite è la vostra immaginazione.

Realizzare un gioco può richiedere ore ed ore di programmazione in linguaggio macchina. E' necessaria inoltre una conoscenza particolareggiata dell'architettura del vostro computer. Data la complessità del lavoro, molte buone idee non sono mai diventate realtà. Ora c'è The Great Arcade Machine, un programma che consente agli utenti di realizzare giochi in maniera interattiva.

Arcade Machine si occupa del lavoro più difficile, e lo esegue alla velocità del linguaggio macchina. Ne risulta che un gioco che potrebbe portare via settimane di lavoro, può venir scritto in un tempo molto breve, dovete solamente imparare ad usare e cambiare i parametri di Arcade Machine secondo regole ben precise.

Il fatto più sorprendente è che Arcade Machine aggiunge ben trentuno istruzioni al

BASIC, permettendovi in tal modo di controllare il vostro gioco tramite il BASIC stesso.

Per iniziare

Il programma 1, "Arcade Machine" ed i programmi 3 e 4 sono interamente scritti in linguaggio macchina; pertanto è indispensabile usare "MLX" per inserirli nel computer. Alla richiesta di istruzioni di MLX rispondete con i valori sotto elencati:

Programma 1:

Indirizzo Iniziale: 0801
Indirizzo Finale: 1BA8

Programma 3:

Indirizzo Iniziale: 30C0
Indirizzo Finale: 3E7F

Programma 4:

Indirizzo Iniziale: 7500
Indirizzo Finale: 7897

Dopo aver inserito tutti i dati, assicuratevi di aver salvato una copia su disco o su nastro prima di abbandonare MLX. Salvate il programma 1 con il nome ARCADE MACHINE, il programma 3 con il nome SPRDATA, ed il programma 4 con il nome PARAMS.

Il programma 2 è scritto in BASIC. Se userete ARCADE MACHINE su nastro, dovete cambiare le linee 60 e 90 del programma 2 come segue:

```
60 PRINT "{HOME}"
  {5 DOWN}{BLU}
  PRINTCHR$(147):CLOAD"A$"
  SPRDATA" A$":CLOAD"A$"
```



```

"PARAMS"AS
90  LOAD"ARCADE MACHINE"
    ,1

```

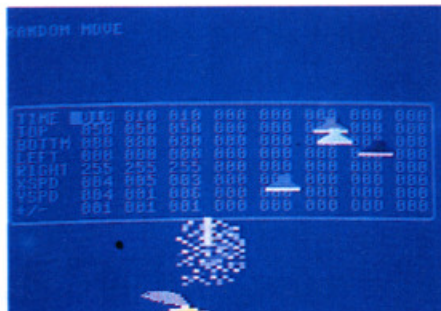
Ora che avete inserito i programmi, siete pronti per vedere cosa è capace di fare Arcade Machine. Inserite un joystick nella porta 2; poi caricate ed azionate BOOT. Usate il joystick per muovere l'elicottero. Potete sparare ai dischi volanti e sganciare bombe sulla nave premendo il pulsante di fuoco, ma attenzione: le collisioni con i dischi sono mortali, e la nave risponde al vostro fuoco.

L'Editor

Arcade Machine possiede un editor incorporato che vi consente di progettare giochi in un batter d'occhio. L'editor si presenta con 18 finestre adibite al controllo del movimento degli sprite, del suono, delle collisioni, del fuoco, e di altri dettagli. L'editor si attiva con l'istruzione WINDOWS. Per il momento fate poco caso agli sprite sullo schermo e digitate WINDOWS. Apparirà così la prima finestra di Arcade Machine, la finestra OPZIONI.

All'interno dell'editor, i tasti f3 ed f7 vi muovono rispettivamente avanti e indietro per le finestre. Il tasto f7 blocca tutti gli sprite e li dispone in basso allo schermo; f5 attiva tutti gli sprite. I tasti corrispondenti ai numeri da 0 a 7 attivano e disattivano gli sprite individualmente. I tasti cursore vi permettono il movimento all'interno delle finestre di Arcade Machine. I tasti + e - incrementano e decrementano il valore selezionato. La barra spaziatrice è attiva su due finestre (SOUND e EXPLODE). Premete la barra spaziatrice su queste finestre per provare un suono o vedere l'esplosione di un suono. Premendo il tasto di RUN/STOP si ritorna al BASIC.

Arcade Machine possiede sei opzioni principali: JOYSTICK, COLLISIONS, REPLACE, BORDERS, RANDOM e FIRE. Ognuna di queste opzioni presenta una o più finestre per cambiare i vari parametri, ma ogni opzione può essere attivata o disattivata da questa finestra. Dando il valore 0 ad un'opzione la si disabilita, il valore 1 la riabilita.



La finestra DEFINE SPRITES vi permette di specificare quali delle sagome dello sprite vanno animate.



La finestra JOYSTICK FIRE vi permette il controllo delle azioni che avvengono alla pressione del pulsante di sparo.

Quando l'opzione JOYSTICK è attiva, voi controllate lo sprite 0 con il JOYSTICK inserito nella porta 2. Quando COLLISIONS è attiva, Arcade Machine esegue l'azione da voi selezionata quando due sprite si toccano. Per esempio Arcade Machine è stato programmato così che la nave esploda quando le bombe la colpiscono.

Quando l'opzione REPLACE ha valore 1, Arcade Machine sostituisce automaticamente gli sprite distrutti. Usate questa proprietà per creare orde infinite di alieni. Gli sprite non usciranno da un'area specifica quando l'opzione BORDERS è abilitata; altrimenti saranno liberi di muoversi ovunque sullo schermo.

L'opzione RANDOM determina il movimento casuale degli sprite. Quando RANDOM è attiva, gli sprite nemici sembrano muoversi in maniera casuale. Anche se il loro movimento appare casuale, un'altra finestra vi consente di decidere con precisione dove devono cambiare movimento, con quale frequenza ed in quale direzione. Quando FIRE è a 0, gli

sprite non possono spararsi l'un l'altro. Quando è regolata a 1, sono liberi di distruggersi a vicenda. Altre finestre controllano la frequenza, direzione e velocità di ogni colpo.

Display degli Sprites

Prima di spostarvi alla prossima finestra, assicuratevi che tutte le opzioni siano attive. Se un'opzione è disattivata, cambiare i parametri in una delle corrispondenti finestre non ha alcun effetto. Questo può confondervi, quando usate l'editor per la prima volta. Ora che tutte le opzioni sono attive, premete f3. La prossima finestra è la DEFINE SPRITES. Questa finestra possiede sette parametri: BLOCK, JIFFS, FRAMES, COLOR, MODE, XSIZE e YSIZE. BLOCK è la locazione di una sezione di memoria di 64 byte che contiene i data che riguardano la forma dello sprite. Visto che il chip VIC-II può indirizzare solo 16K di memoria alla volta, ci sono 256 blocchi di memoria disponibili (da 0 a 255) per i data degli sprite. BLOCK, JIFFS e FRAMES riguardano l'animazione degli sprite. Per ottenere l'animazione, uno sprite compie un ciclo attraverso parecchie forme differenti. I DATA relativi alla forma di uno sprite vengono depositati in blocchi di memoria consecutivi. BLOCK è la locazione della prima definizione di forma. FRAMES è il numero di definizioni attraverso le quali lo sprite deve compiere il ciclo, e JIFFS è l'intervallo in jiffies tra un cambiamento e l'altro (un jiffy corrisponde ad 1/60 di secondo). Uno sprite con JIFFS regolato a 0 non compie alcun ciclo. Una volta che questi parametri sono stati regolati, Arcade Machine compie automaticamente l'animazione degli sprite per voi. Per esaminare più da vicino l'animazione, premete f7 per disattivare gli sprite. Regolate il JIFFS di uno dei dischi a 60 (un secondo tra un cambiamento e l'altro) e lo osserverete compiere lentamente un ciclo attraverso tre diverse forme. Regolate il colore di uno sprite con il parametro COLOR. MODE trasferisce lo sprite dentro e fuori il multicolor mode. Per raddoppiare la dimensione dello sprite orizzontalmente date a XSIZE o YSIZE, rispettivamente il valore 1. Arcade Machine non possiede

un editor di sprite incorporato. Pertanto, usate uno dei molti editor di sprite disponibili per creare i vostri sprite. Più avanti, avrete modo di vedere come si possono caricare i data della forma di uno sprite in Arcade Machine usando le istruzioni DLOAD o CLOAD.

Suoni

Un'appropriato uso del suono è molto importante, in qualsiasi gioco. Azioni come lo sganciamento delle bombe o il lancio dei missili, il fuoco delle mitragliatrici e le esplosioni, sono tutte enfatizzate da un appropriato uso del suono. Arcade Machine vi consente di generare suoni interessanti con la prossima finestra che esamineremo: la finestra SOUND.

Arcade Machine vi permette di definire un massimo di 8 suoni diversi. Per cambiare un suono, muovete il cursore su una colonna che concerne il suono e inserite un numero di WAVE (onda). I numeri dell'onda sono 1, 2, 3, 4, e a questi corrispondono rispettivamente onde con le forme: triangle (triangolo), sawtooth (dente di sega), square (quadrato) e noise (rumore). Sono disponibili 96 note differenti (0-95) per otto ottave (vedi Appendice E nella Guida di Riferimento del Programmatore). NOTE 1 è la nota d'inizio del suono e NOTE 2 è quella finale, TEMPO riguarda la velocità con la quale il suono si alza e decade. Se il valore di TEMPO è uguale a 0, si ha una nota ferma. L'involucro del suono è regolato dai parametri ATTACK, DECAY, SUSTAIN e RELEASE. TIME determina la durata del suono. Quando TIME è uguale a 0, il suono non termina mai. Per ascoltare il suono, premete la barra spaziatrice.

Arcade Machine usa la voce 1 per il fuoco del vostro sprite e la voce 2 per altri suoni. La voce 3 è usata per generare note casuali. Dato che ci sono solo due voci disponibili, possono venire a crearsi dei conflitti. Se una voce viene usata per produrre due suoni contemporaneamente, uno taglia fuori l'altro.

Controlli joystick

Nei giochi come Space Invaders, il vostro sprite si può muovere solo a destra e a sinistra. Nei giochi come Pac-Man, non è

possibile fermare il vostro sprite centrando il joystick. In altri giochi lo sprite si ferma quando centrate il joystick, ma potete sparare solo nella direzione verso la quale siete rivolti. Arcade Machine ha due finestre per il controllo di questi fattori: JOYSTICK e JOYSTICK FIRE.

La finestra JOYSTICK controlla i movimenti del vostro sprite. Presente cinque direzioni: CNTR, UP, DOWN, LEFT e RGHT. Il parametro ENABLE determina in quali di queste direzioni si può muovere il vostro sprite (CNTR controlla che lo sprite si fermi quando il joystick viene centrato). BLOCK, JIFFS e FRMES controllano l'animazione dello sprite in movimento. SPDEN stabilisce se lo sprite deve cambiare velocità quando viene mosso il joystick. Notate la differenza tra SPDEN ed ENABLE: se SPDEN è uguale a 0, lo sprite può girarsi e sparare, ma non può muoversi. Se ENABLE è uguale a 0, lo sprite non può assolutamente né girare, né sparare, né muoversi. XSPEED e YSPEED regolano la velocità dello sprite quando si muove nell'una e nell'altra direzione.

I parametri joystick possono essere cambiati interattivamente, ma dovete muovere il joystick per vedere la differenza. Se il joystick è diretto in alto, per esempio, dovete muoverlo in un'altra direzione e poi muoverlo di nuovo verso l'alto per vedere il cambiamento. Se il joystick è centrato, dovete muoverlo e poi centrarlo nuovamente per vedere cosa è cambiato. La finestra JOYSTICK FIRE controlla il fuoco dello sprite.

I parametri BLOCK, JIFFS e FRMES determinano l'animazione dello sparo dello sprite. COLOR, MODE, XSIZE e YSIZE regolano il colore e le dimensioni del colpo. XSPEED ed YSPEED la velocità. TIME è il tempo, in jiffy, che deve trascorrere prima che possiate sparare di nuovo. Se volete simulare un ritmo di fuoco di un colpo per secondo, regolate time a 60. Per simulare una mitragliatrice, regolatelo a 5. Per disattivare il fuoco in una certa direzione, date al TIME proprio di quella direzione il valore 0. XOFF e YOFF indicano la distanza verticale ed orizzontale del punto di partenza del colpo dalla nave.

SOUND è il numero (1-8) del suono che produce l'astronave al momento dello sparo. Se mettete SOUND a 0, la nave non fa rumore quando spara.

Esplosioni !

Le prossime sei finestre -EXPLODE, ON COLLISION, SOUND ON COLLISION, POINTS ON COLLISION, ON SPRITE/DATA COLLISIONS, e REPLACE- hanno tutte qualcosa in comune. Queste finestre controllano le collisioni tra gli sprite e i data; determinano pure forma e suono delle esplosioni.

La finestra EXPLODE controlla il rumore e l'immagine di un'esplosione. BLOCK, JIFFS e FRMES controllano l'animazione dell'esplosione. COLOR, MODE, XSIZE e YSIZE ne controllano colore e dimensioni. SOUND determina quale suono genera l'esplosione quando si verifica. Potete provare un'esplosione seguendo questo procedimento: premete f7 per disattivare gli sprite, posizionate il cursore sulla colonna dello sprite che desiderate vedere esplodere e premete la barra spaziatrice.

La finestra ON COLLISION comunica ad Arcade Machine quale azione portare a termine quando lo sprite nella colonna di sinistra si scontra con lo sprite nella fila in alto. I valori per queste azioni sono mostrati nella tabella 1.

DIE è una sequenza di azioni comune a molti giochi: ferma tutti gli sprite, fa esplodere lo sprite, fa una pausa, e poi disattiva tutti gli sprite.

La finestra SOUND ON COLLISION determina quale suono si deve udire quando due sprite collidono. Un valore di 0 vuol dire che non si sente alcun suono. La finestra POINTS ON COLLISION controlla se il punteggio deve aumentare quando si verifica uno scontro. La finestra ON SPRITE/DATA COLLISIONS è esattamente uguale alla finestra ON COLLISION, solo che controlla le collisioni tra sprite e data di sfondo.

La finestra REPLACE controlla come gli sprite debbono riapparire. DELAY è il tempo che trascorre prima che lo sprite compaia, e XPOSITION e YPOSITION sono la colonna e la fila della base dove ricompare lo sprite. +RND è il più alto

numero casuale che può venir aggiunto alla posizione della base dello sprite. Per esempio, se volete che uno sprite ricompaia ovunque tra Y=50 e Y=200, regolate la sua YPOSITION a 50 e il suo +RND a 150. Se volete che uno sprite ricompaia sempre nella stessa fila o colonna, date a +RND un valore 0. XSPD e YSPD sono le velocità iniziali dello sprite X ed Y. L'ultimo parametro, +/- è usato per cambiare la velocità dello sprite. I valori di +/- sono mostrati nella tabella 2.

Per provare l'effetto reciproco dei parametri REPLACE, premete f5 per disattivare tutti gli sprite. Ciò costringe Arcade Machine a sostituirli.

Definizione dei bordi

Le prossime finestre che prenderemo in esame vengono usate per determinare i limiti del movimento degli sprite. La finestra BORDERS pone i limiti ad ogni sprite. La finestra ACTION AT BORDERS vi permette di decidere cosa deve succedere quando uno sprite tocca uno dei limiti del suo campo di movimento, e SOUND AT BORDERS quale suono viene prodotto quando questo si verifica. POINTS AT BORDERS regola il punteggio extra dovuto a questa eventualità. La finestra BORDERS deve essere usata per definire una finestra all'interno della quale ogni sprite può muoversi. Per esempio, se intendete limitare il movimento di un disco alla metà superiore dello schermo, assegnate al parametro BOTTM il valore di 99. I bordi a destra e a sinistra sono un poco complicati. Dato che un byte contiene solo 256 valori, i numeri dei bordi sono moltiplicati X2 per ottenere le coordinate effettive dello schermo. Così dovreste limitare i bordi laterali tra 0 e 159 (assegnare al bordo destro o sinistro un numero più alto di 159 può causare lo sconfinamento dello sprite nel lato sinistro dello schermo inaspettatamente).

Le finestre ACTION AT BORDERS e SOUND AT BORDERS controllano l'azione e il suono che si producono quando uno sprite tocca un bordo. ACTION AT BORDERS esige un numero tra 0 e 9 che corrisponde alle azioni mostrate nella tabella 1. SOUND AT BORDERS richiede il numero del suono che genera lo sprite

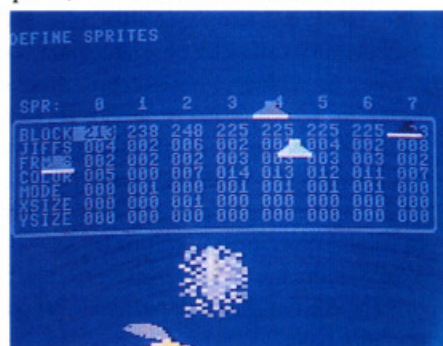
quando viene a contatto con un bordo (10 significa che non viene prodotto alcun suono). Se volete che aumenti il punteggio quando uno sprite tocca il bordo, inserite il numero dei punti nella finestra POINTS AT BORDER. Gran parte dei valori di questa finestra dovrebbero essere 0, non avrebbe senso incrementare il punteggio quando un disco tocca il suo bordo sinistro o destro.

Movimento casuale

La finestra ENABLE RANDOM MOVE ha la funzione di selezionare quali sprite possono cambiare movimento casualmente. La finestra RANDOM MOVE controlla la casualità del movimento di uno sprite. Ogni colonna in questa finestra definisce un'area dello schermo. Se uno sprite è all'interno di una di queste aree, o più d'una, il programma cambia la sua velocità secondo le vostre indicazioni. Se lo sprite è all'interno di più aree delimitate, reagisce solo ad una di esse (scelta casualmente). Se lo sprite non entra in una delle aree predefinite, la sua velocità resta invariata.

Nota: il movimento casuale degli sprite 1 e 7 viene generalmente disattivato. Se non fosse così, la linea di fuoco del vostro sprite o degli sprite nemici si muoverebbe casualmente.

Il primo parametro, TIME, indica il tempo (in jiffies) che deve passare prima che la velocità cambi nuovamente. Una colonna viene esclusa se il suo TIME corrisponde a 0. TOP, BOTTM, LEFT e RIGHT definiscono una regione rettangolare dello schermo. Se Arcade Machine decide di cambiare il movimento di uno sprite, XSPD e YSPD saranno i nuovi valori per la velocità orizzontale e verticale. I valori per +/- sono elencati nella tabella 2.



Fuoco nemico

Tutti i giochi d'azione che si rispettino hanno delle navi nemiche che rispondono al vostro fuoco. La finestra ENEMY FIRE vi consente di determinare la capacità di risposta di tali navi. RATE controlla la frequenza dei colpi nemici (se volete che non sparino del tutto, mettete RATE a 0). DIRECTION è la direzione nella quale le navi nemiche possono far fuoco. Assegnando al parametro DIRECTION il valore 0, il nemico sparerà in alto; con il valore 1 in basso; DIRECTION pari a 2 vuol dire che sparirà a sinistra; e 3 significa che sparirà a destra. Quando l'AIM di uno sprite nemico ha valore 1, questo sparirà solo quando sarà allineato con il vostro sprite. Quando AIM è uguale a 0, lo sprite spara casualmente. SPEED è la velocità alla quale viaggiano i colpi, e SOUND il suono che producono gli spari. Come nel caso della finestra JOYSTICK FIRE, XOFFS ed YOFFS indicano la distanza verticale ed orizzontale che deve intercorrere tra la nave ed il punto di partenza del colpo. Lo sprite 7 è adibito allo sparo nemico. Deve essere spento perché funzioni ENEMY FIRE. Dovreste anche regolare Arcade Machine perché esso venga disattivato quando raggiunge i propri bordi (date ai suoi parametri di ACTION AT BORDER valore 1).

Un buon modo per provare come i parametri di ENEMY FIRE funzionano interattivamente, è quello di disattivare COLLISIONS, regolare RATE al massimo (255), premere f7 per mettere tutti gli sprite in basso, premere 7 per disabilitare lo sprite 7 e poi posizionare l'elicottero di fronte agli sprite nemici.

Istruzioni BASIC

Arcade Machine fornisce una serie di istruzioni BASIC aggiuntive, che vi consentono di modificare i parametri senza fare uso dell'editor. Nelle istruzioni seguenti, le parentesi quadre indicano che il parametro è facoltativo.

SPRITE [numero di sprite]. L'istruzione SPRITE attiva un particolare sprite. Se si digita SPRITE senza il numero di sprite, si attivano tutti gli sprite.

OFF [numero di sprite]. Questa istruzione disattiva uno sprite. OFF senza nume-

ro di sprite disattiva tutti gli sprite.

HALT [numero di sprite]. L'istruzione HALT blocca uno sprite. HALT senza numero di sprite ferma tutti gli sprite.

MOVE numero di sprite, x, y. L'istruzione MOVE cambia la velocità di X e di Y dello sprite.

XMOVE numero di sprite, x. Cambia la velocità X dello sprite.

YMOVE numero di sprite, y. Cambia la velocità Y dello sprite.

COLOR numero di sprite, c. Cambia il colore di uno sprite in Hi-res.

MULTIC c1, c2. Cambia i colori di uno sprite in multicolor.

MODE numero di sprite, n. Decide se uno sprite è in Hi-res o in multicolor mode. Quando n è 0, lo sprite è in Hi-res, quando n è 1, lo sprite va in multicolor.

PRIORITY numero di sprite, n. Determina la priorità di uno sprite. Quando n è 0 lo sprite ha la priorità rispetto ai dati dello sfondo, quando è 1, i dati di sfondo hanno la priorità.

ANIM numero di sprite, block, frames, jiffies. Attiva l'animazione di uno sprite.

REL numero di sprite, x, y. Cambia le coordinate X ed Y di uno sprite.

XREL numero di sprite, x. Cambia la coordinata x di uno sprite.

YREL numero di sprite, y. Cambia la coordinata y di uno sprite.

SIZE numero di sprite, xsize, ysize. Determina la dimensione X ed Y di uno sprite. Quando xsize e ysize sono 0, lo sprite è normale. Un xsize di 1 ingrandisce lo sprite orizzontalmente, un ysize di 1 lo espande verticalmente.

SOUND numero del suono [nota]. Attiva uno degli otto suoni. Il valore 'nota' è facoltativo.

BLAST numero di sprite. Fa esplodere uno sprite (prima dovete regolare i parametri della finestra EXPLODE).

DLOAD "nome del file". Carica un parametro o i dati relativi alla forma di uno sprite dal disco.

DSAVE "nome del file". Salva un parametro o i dati relativi alla forma di uno sprite su disco.

CLOAD "nome del file". Carica un parametro o i dati relativi alla forma di uno sprite da nastro.

CSAVE "nome del file". Salva un para-

metro o i dati relativi alla forma di uno sprite su nastro.

SCREEN numero. Determina la locazione dello schermo in memoria. Il numero è la locazione di start dello schermo divisa per 256. Per esempio, per cambiare nello schermo usato da Mob Maker, digitate SCREEN 128 (32768/256=128). Per ritornare allo schermo normale digitate SCREEN 4.

ENABLE [joystick, collisions, replace, borders, random, fire]. Attiva le opzioni. Digitando solo ENABLE si attivano tutte le opzioni.

DISABLE [joystick, collisions, replace, borders, random, fire]. Disattiva le opzioni. Digitando solo DISABLE si disattivano tutte le opzioni.

Mettete tutto insieme

I giochi in BASIC basati sugli sprite sono facili da realizzare con Arcade Machine. Il programma BOOT mostra ciò che è necessario per iniziare, caricate Arcade Machine, un file di parametri, ed un file di DATA degli sprite. La sola cosa che dovete aggiungere a BOOT è un loop che mostri il punteggio. Il loop deve continuare fino a quando il numero delle vite rimanenti è uguale a 0; a questo punto potrebbe, per esempio, disattivare tutti gli sprite, porre la domanda VUOI GIOCAR ANCORA? ed aspettare una risposta. Se la risposta è no, il gioco finisce, altrimenti il punteggio si azzera, il numero delle vite diventa quello iniziale, ed il loop ricomincia. Ecco gli indirizzi dei quali avete bisogno per fare il PEEK o POKE dal BASIC:

700-70 posizione X degli sprite da 0 a 7
708-715 posizione y degli sprite da 0 a 7
716-723 vite degli sprite da 0 a 7

724-725 punteggio, SCORE (low byte/high byte - byte basso/byte alto)

726 specchio dello sprite/collisione-sprite registro d'individuazione \$ D01E

727 specchio dello sprite/collisione-data registro d'individuazione \$ D01F

Ricordatevi di fare il POKE 0 in una "locazione-specchio" dopo averla letta.

Tabella 1: Collisioni

Valore	Azione
0	Ignora
1	Off
2	Esplode
3	Muore
4	Stop (XSpeed=0; YSpeed=0)
5	Cade (XSpeed=0; YSpeed=16)
6	RimbalzoY (YSpeed=-YSpeed)
7	RimbalzoX (XSpeed=-XSpeed)
8	Scorrimento Y (YSpeed=0)
9	Scorrimento X (XSpeed=0)

Tabella 2: Variazioni di velocità

Valore	Significato
0	Non si varia la velocità
1 o -1	La velocità X degli sprite varia in maniera casuale
2 o -2	La velocità Y degli sprite varia in maniera casuale
3 o -3	La velocità X e la velocità Y degli sprite variano simultaneamente in maniera casuale

Home Recording Studio

Una realtà per tutti con C64 e programmi sequencer MIDI

Aldo Laus

L'applicazione più interessante e creativa per un computer nel mondo MIDI pensiamo sia quella del suo utilizzo come sequencer in uno studio di registrazione MIDI. Anche per una macchina semplice come il C 64 l'impiego in questo senso può essere pieno di soddisfazioni grazie all'altissimo livello di sofisticazione raggiunto da alcuni pacchetti software che indicheremo di seguito. Prima però di entrare nei dettagli specifici dei programmi sequencer per il C64, vogliamo, a beneficio dei novizi, parlare un pò in generale del tema di questa puntata. Nell'era A.M. (Ante MIDI) ovvero prima del 1983, chi voleva realizzare sia a scopo professionale che amatoriale delle registrazioni musicali multipista, utilizzando come sorgente di suono diverse voci del proprio sintetizzatore, doveva investire una discreta somma per un registratore multipista e doveva poi sudare mica male per eseguire una parte dopo l'altra alla perfezione, incidendole progressivamente sul nastro, pena il dover rifare completamente l'ultima parte finchè non fosse venuta come si deve. Solo alla fine dell'incisione comunque, dopo aver giudicato l'insieme, si poteva stabilire se l'opera era a posto oppure rendersi conto che il volume ad esempio degli strings era troppo alto o, invece del clarino sarebbe stato più appropriato un flauto per una certa parte. A questo punto, se la parte strumentale da rifare era su una pista individuale, poco



male, ma se avendo a disposizione poche piste si era già ricorsi a delle sovraincisioni beh, forse avreste rimandato tutto il lavoro al giorno seguente. Nella migliore delle ipotesi, anche se la vostra esecuzione fosse risultata perfetta e così pure il messaggio e il colore dei toni, la registrazione eseguita in questo modo "tradizionale" avrebbe potuto soffrire di altre limitazioni che, anche se di fatto non fossero risultate immediatamente, probabilmente in qualche occasione sarebbero emerse inesorabilmente. Citiamo un paio di esempi che possono riflettere esigenze pratiche tipiche.

- Una volta inciso il brano sul nastro non è possibile variarne la velocità di esecuzione, mentre a volte può essere interessante provare queste variazioni, se non indispensabile quando dovete comporre un commento musicale per un pezzo video di durata prefissata.

- Non si può trasporre di tonalità una base incisa su nastro. Se avete questa esigenza occorre rifare l'esecuzione del brano nella tonalità desiderata.

La soluzione MIDI

Con un sequencer MIDI e ovviamente con un sintetizzatore MIDI come sorgente sonora, si apre un nuovo modo di concepire la registrazione in cui, il passaggio del suono sul nastro viene realizzato solo come atto finale, dopo aver fatto tutte le esecuzioni e le eventuali correzioni delle varie parti fino ad essere soddisfatti del risultato finale sulle tracce del sequencer che, se vogliamo, può essere considerato un vero e proprio registratore multipista. Vediamo quindi come funziona il meccanismo. Tutti sanno ormai che quando eseguiamo un pezzo musicale su di un sintetizzatore MIDI, tutte le azioni relative vengono codificate istantaneamente in messaggi digitali MIDI che sono inviati alla presa MIDI OUT dello strumento. I dati corrispondenti ai messaggi giungono al nostro computer attraverso la presa MIDI IN dell'interfaccia MIDI e qui vengono opportunamente gestiti da un programma sequencer che li registra su una delle tracce di cui dispone. Fisicamente una traccia è costituita da un certo

numero di memorie RAM del computer, in funzione della lunghezza del brano registrato. Terminata la registrazione, dando un comando "PLAY" al sequencer, le memorie RAM interessate vengono lette dal programma e quindi il contenuto della traccia può essere inviato (attraverso l'interfaccia MIDI) allo strumento MIDI che, interpretando i dati, ne esegue il brano. Ovviamente durante l'esecuzione avremo collegato al nostro sintetizzatore un registratore magnetico in modo da immortalare il prodotto della nostra creatività. Questo è il caso ovviamente più semplice, nell'ipotesi che l'esecuzione sia avvenuta subito perfettamente. Se invece qualcosa va modificato, vediamo cosa si può fare dopo aver registrato la traccia nel sequencer. Come abbiamo visto sopra, tutte le informazioni relative alla traccia sono presenti nella RAM del computer perciò tramite opportuni comandi del programma si possono richiamare e modificare con estrema facilità servendosi di schermate video. A seconda del programma potrete avere un semplice elenco degli eventi MIDI presenti nella traccia, ovvero la rappresentazione alfanumerica del file dei dati, oppure, nei programmi più sofisticati, una rappresentazione grafica degli eventi. In entrambi i casi comunque si può facilmente modificare qualunque dettaglio della rappresentazione corrispondente ai parametri della nostra esecuzione. Questa funzione del sequencer è chiamata Editing. Un sequencer quindi è tanto più potente quanto più numerose e sofisticate sono le opzioni di Editing. Vediamone le più importanti.

- Trasposizione: consente di traslare di tonalità tutte le note contenute nella traccia.
- Cambio del canale MIDI: consente di modificare il dato relativo al numero del canale MIDI a tutti gli eventi presenti nella traccia caratterizzabili col canale. Ciò consente di indirizzare in ascolto il contenuto di una traccia su un canale (e quindi su uno strumento) diverso da quello registrato in origine.
- Modifica del livello di volume di una traccia: si può ottenere l'equivalente della regolazione del volume su di un mixer audio.

- Sostituzione di una parte della traccia con un'altra parte preparata appositamente in un buffer o su un'altra traccia.
- Modifica del tempo di esecuzione della traccia: comprime o espande la durata della traccia variandone il tempo di esecuzione.
- Aggiunta di effetto eco: si possono dosare il numero e i volumi dei rimbalzi.
- Possibilità di copiare una traccia da un'altra. Oltre alle funzioni di editing vere e proprie, ci sono anche diverse modalità di predisposizione per il funzionamento del sequencer che si rivelano molto utili per raggiungere facilmente il risultato finale fra cui:
 - Variazione della divisione del tempo (4/4, 3/4, 6/8 ecc.)
 - Inserzione di un metronomo audio durante la registrazione, con partenza anticipata rispetto alla stessa di un numero definibile di battute
 - Filtro di determinati eventi MIDI per risparmiare Kbit di preziosa memoria RAM per immagazzinare solo i dati più salienti da noi prescelti
 - Scelta fra sincronizzazione interna o esterna (ad esempio da parte di una drum machine o di un altro sequencer)
 - Predisposizione del "Quantize" cioè del passo di risoluzione minimo della battuta che in alcuni programmi definisce anche il passo di autocorrezione (per ottenere una perfetta quadratura nel caso aveste la mano incerta nell'esecuzione). Quanto

abbiamo descritto per una traccia ovviamente vale per tutte le tracce che possono essere presenti contemporaneamente sul sequencer e che nell'insieme consentono, supportando in uscita il suono di un diverso strumento musicale generato dal sintetizzatore, di realizzare un brano musicale a più parti. Le esecuzioni, dopo essere state messe a punto secondo i nostri gusti o le esigenze, possono anche essere salvate su disco sotto forma di files, pronte quindi per successivi ascolti o per ulteriori elaborazioni.

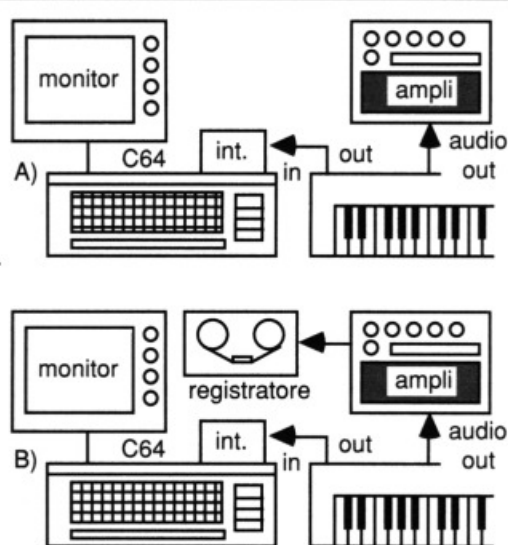
MIDI home recording

Per definire cosa serve per realizzare uno studio di registrazione MIDIsalingo, conviene esaminare il tema da tre punti di vista:

1 - Computer, per far funzionare il quale occorre ovviamente un buon programma sequencer (vedi il riquadro con alcuni esempi) in grado di soddisfare le vostre aspettative. E' opzionale solo la stampante in quanto solo se disponete di un programma di notazione musicale potete, se lo ritenete necessario, stampare lo spartito delle vostre esecuzioni.

2- Gli strumenti MIDI ideali per realizzare registrazioni multiparti sono quelli in cui è possibile suddividere in gruppi (in modo fisso o dinamico) i generatori di note a disposizione ed assegnare, ad ogni gruppo risultante sia un timbro strumentale che un canale MIDI in ricezione diversi.

Durante la registrazione (A) i dati MIDI relativi all'esecuzione passano dalla presa out dello strumento alla presa in dell'interfaccia. Questa li invia al programma sequencer che li inserisce in una traccia. L'una dopo l'altra vengono così registrate le varie tracce del brano. Durante l'ascolto (B) il programma sequencer restituisce i dati presenti nelle tracce registrate in precedenza inviandoli, tramite l'interfaccia, al sintetizzatore che effettua l'esecuzione di tutte le parti contemporaneamente dai vari canali MIDI interessati. Si realizza quindi l'incisione su nastro



Oggi sintetizzatori o expanders con queste caratteristiche sono molto diffusi ed i prezzi sono sempre più accessibili quindi la possibilità di accedere a questo tipo di tecnica si sta aprendo a più vaste schiere di appassionati. Quanto sopra non esclude la possibilità di utilizzare gli strumenti MIDI che lavorano solo su un canale per realizzare registrazioni polifoniche tramite un sequencer. Il fatto che ci fa favorire il primo metodo è che usando strumenti del secondo tipo il processo diventa più macchinoso, occorre infatti interporre fra computer e registratore un altro accessorio (il convertitore di sincronismo FSK) e, fatto più grave economicamente, è indispensabile dotarsi di un registratore audio multitraccia.

3 - Se disponiamo di una tastiera multitimbrica su diversi canali MIDI come descritto al punto 2, sarà invece sufficiente disporre di un comune registratore stereo HI-FI casalingo per ottenere buoni risultati. Se poi il vostro budget vi consente registratori multitraccia, mixers ed effetti speciali (oggi anche questi possono essere pilotati via MIDI tramite una traccia del sequencer), non possiamo che complimentarci con voi ed augurarvi delle buone registrazioni.

La registrazione sui sequencer MIDI

Salvo rare eccezioni, nel mondo dei sequencer si lavora in termini di SONG & PATTERN, questi ultimi noti anche col termine sequenze. La SONG può essere il prodotto finale che rappresenta il brano musicale completo che viene eseguito dal sequencer. La SONG è costituita dalla sequenza di un certo numero di pattern che avrete elaborato e numerato separatamente e che, con opportuni comandi, avrete deciso di cucire insieme in un certo ordine, anche magari ripetendone alcuni, secondo le esigenze del vostro brano. Scendendo ancora di più nell'analisi, un pattern si può definire con i seguenti elementi:

- numero di battute che lo costituiscono
- numero di tracce (una per ogni voce strumentale che vogliamo produrre).

Chiarito il meccanismo operativo del sequencer, vediamo il comportamento che deve attuare il musicista per trarre il

maggior vantaggio dalla macchina. Un brano musicale è generalmente divisibile in parti (introduzione, inciso, ritornello ecc.) ed è facile che alcune di queste si ripetano nell'ambito del brano stesso. Stabiliamo, per intenderci, che una parte musicale corrisponde ad un pattern del sequencer.

Al musicista converrà quindi analizzare il brano su cui intende lavorare ed individuare se è possibile suddividerlo in parti ripetibili e parti singole. Se ciò risulta possibile, non è quindi necessario incidere l'intero brano, ma solo una volta ciascuna parte (sia le singole che le ripetiti-

ve) in altrettanti pattern. Ecco quindi che, specificando l'ordine in cui i pattern dovranno essere eseguiti, otterremo di fatto l'esecuzione della SONG da parte del sequencer. E' consigliabile quindi un lavoro di analisi del materiale musicale prima di usare il sequencer per ottimizzare il lavoro. All'uopo basterà semplicemente compilare una tabella in cui evidenziare per il brano in oggetto:

- il numero di patterns da incidere
- il loro ordine per costituire la song
- il numero di battute per ogni pattern
- il numero di tracce che costituiscono ogni pattern.

Nello specchietto qui rappresentato, elenchiamo alcuni tra i più famosi programmi sequencer MIDI per C64/128 nonché i nominativi del produttore o del distributore italiano



TITOLO: Pro 16
PRODUTTORE: Steinberg
DISTRIBUTORE: Midiware srl



TITOLO: Recording Studio
PRODUTTORE: Jellinghouse



TITOLO: Supertrack
PRODUTTORE: C-Lab



TITOLO: Live Sequencer
PRODUTTORE: SIEL

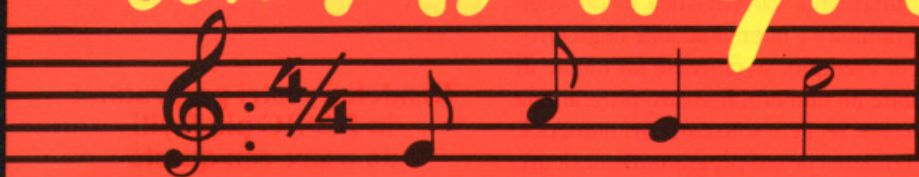


TITOLO: Instant Music
PRODUTTORE: Electronic Arts

GRUPPO EDITORIALE
JACKSON



ED E' SUBITO MUSICA
CON **AMIGA**



**CORSO RAPIDO,
FACILE, COMPLETO**

NUMERO UNICO

A sole £ 19.000
Frs. 28,50

MUSICA
con
AMIGA



GRUPPO EDITORIALE
JACKSON

IN EDICOLA

La potenza del BASIC

Un'importante caratteristica di un buon programma è l'intercettazione degli errori. Per esempio, se chiedete all'utente di digitare un numero, il vostro programma dovrebbe leggere solo caratteri numerici. Per rendere i propri programmi più sicuri, molti programmatori disattivano il tasto STOP per evitare che l'utente fermi il programma inopportunitamente (ad esempio durante il Save o Load). Un'idea ancora migliore è quella di premunire il computer così che quando si preme il tasto STOP viene eseguita una determinata azione.

"On Break Goto" è stato progettato per questo scopo. Quando viene attivata, la versione per il C64 di On Break Goto verifica continuamente che non venga premuto il tasto STOP. Quando questo accade, On Break Goto salta automaticamente ad una specifica linea del vostro programma.

La versione del C128 di On Break Goto è ancora più potente. Può eseguire una serie di istruzioni BASIC, come saltare ad una subroutine, fare il display di una directory, o persino far funzionare un altro programma.

Per iniziare

Entrambe le versioni di On Break Goto sono scritte in linguaggio macchina, ma sono qui listate come caricatori BASIC. Coloro che posseggono un Commodore 64 devono digitare il Programma 1.

Questo programma fa il POKE di On Break Goto nella memoria e lo attiva. Entra poi in un loop infinito alla linea 110. Premete il tasto STOP per interrompere il loop. Normalmente, ciò vi riporterebbe al BASIC, ma non quando On Break Goto è attivo. Invece On Break Goto causa una deviazione alla linea 280.

Gli utenti di Commodore 128 devono battere i Programmi 2-5. Assicuratevi di essere nel 128 mode quando battete questi programmi. Il programma 2, Break Maker, crea la versione per il 128 di On Break Goto: ON BREAK/CMD. I programmi 3-

**Rendete i vostri programmi
in BASIC più facili
da usare con questa
utility in l.m.
Anziché disattivare
lo STOP, potrete
aggiungere l'istruzione
ON BREAK GOTO
al vostro C64 e istruzioni
ancora più potenti al C128.**

Matthew P. Gilbert

5 contentgono un demo che vi mostra come potete usare la versione di On Break Goto per il 128. Battete i programmi e salvateli su disco con i nomi DEMO (per il Programma 3), SUB1 (per il Programma 4) e SUB2 (per il Programma 5). Dopo aver inserito questi programmi, fate girare Break Maker per creare ON BREAK/CMD. La versione del 128 è ricollocabile, così Break Maker chiede un indirizzo di start. Premete RETURN per ottenere l'indirizzo di default 2816. Dopo che ON BREAK/CMD è stato creato, caricate DEMO e battete RUN per vedere la versione di On Break Goto in azione.

Come funziona

Naturalmente, quando viene premuto il tasto STOP, il C64 ed il C128 saltano attraverso un vettore a \$328-\$329. Quando viene attivato per la prima volta, On Break Goto intercetta questo vettore e posiziona il proprio indirizzo in queste locazioni. Dato che ora il vettore è diretto a On Break Goto questo viene richiamato ogniqualvolta il tasto STOP viene premuto. Usando la tecnica della tastiera dinamica, la versione del C64 di On Break Goto prima rassetta lo schermo e poi stampa un GOTO xxxx, dove xxxx è il numero della linea di destinazione. Il 128

usa la stessa tecnica di base, ma lo potete personalizzare per eseguire una varietà di istruzioni BASIC. Personalizzare la versione del C64 è facile: basta cambiare l'LN\$ alla linea 60 nel numero della linea alla quale desiderate che salti quando viene premuto il tasto STOP. Potete includere On Break Goto in tutti i vostri programmi in BASIC. Fate il POKE dei dati nella memoria vicina all'inizio del vostro programma e digitare SYS 49152 per attivare On Break Goto. Per disattivarlo, digitate nuovamente SYS 49152. Per personalizzare la versione del 128, cambiate A\$ alla linea 50 nel comando che volete che venga eseguito quando viene premuto il tasto STOP. Potete far eseguire il comando che desiderate, ma siete limitati ad un massimo di 95 caratteri. L'ultimo comando dovrebbe cancellare lo schermo. Poi, cambiate il nome del file alla linea 110; poi fate girare il programma per creare una nuova versione di On Break Goto. Il vostro programma in BASIC può "BLOADare" questo file a necessità. Dato che la versione del 128 di On Break Goto è ricollocabile, dovete essere voi ad installare il vettore STOP. Fate il POKE dell'indirizzo di start di On Break Goto nelle locazioni di memoria 808 e 809 sotto forma di byte bassi ed alti. Le seguenti linee permettono ciò:

```
10 SA=2816:REM INDIRIZZO
    INIZIALE
20 HI=INT(SA/256):LO=SA-
    (HI*256):POKE 808,LO
    :POKE 808,HI
```

Inoltre, il vostro programma deve disattivare la combinazione RUN/STOP-RESTORE. Il comando POKE 792,98 provvederà a questo. Per ripristinare tutto alla normalità, inserite la linea seguente:

```
POKE 808,110:POKE 809,246:
POKE 792,64
```

Vedere il listato a pagina 61

L'angolo GEOS

Il deskTop GEOS è una potente applicazione; è l'interfaccia all'intera famiglia GEOS. Purtroppo però, manca di una caratteristica: lo screen dump. "Screen Dumper" è uno screen dump in linguaggio macchina per GEOS, nella forma di un accessorio da tavolo. Funziona in congiunzione con deskTop e con la maggior parte delle applicazioni che prevedono il supporto di accessori desk. Visto che Screen Dumper utilizza il sistema di chiamata dello standard GEOS, dovrebbe funzionare con qualsiasi stampante per la quale possedete un disk driver.

Come inserire il programma

Visto che Screen Dumper è un programma scritto in linguaggio macchina, dovreste inserirlo con "MLX", il programma d'inserimento del linguaggio macchina lo trovate alla fine di questa rivista (gli utenti di Commodore 128 devono mettere il proprio computer nel 64 mode quando digitano il programma). Quando l'MLX sarà pronto ed aspetterà una vostra istruzione rispondete con i seguenti valori:

INDIRIZZO INIZIALE: 16A5
INDIRIZZO FINALE: 1790

Inserite ora i dati di Screen Dumper, Programma 1. Quando avete finito salvate delle copie del programma sotto il nome di SCREEN DUMPER su almeno 2 dischi. Assicuratevi che uno dei dischi sia un disco di lavoro GEOS. Uno di questi files verrà modificato dal Programma 2; l'altro è un back-up, in caso qualcosa non andasse per il verso giusto.

Ora battete il Programma 2: "BSD" (anche questa volta, gli utenti di C 128 devono posizionare il computer nel 64 mode). Salvate copie di BSD sui dischi che contengono Screen Dumper.

BSD serve per inserire Screen Dumper in un accessorio desk GEOS. Questo è possibile poichè scrive direttamente sulla directory del vostro disco. Se avete fatto

qualche errore durante la battitura di BSD, può succedere che questo distrugga la directory del vostro disco. Perciò fate attenzione e assicuratevi che funzioni correttamente. Quando metterete in azione BSD per la prima volta, memorizzatelo insieme ad una copia di Screen Dumper su un disco a parte, dove non vi sono altri programmi. In questo modo, se qualcosa dovesse andare "storto", non perderete inutilmente dei file.

Mettete in funzione il Programma 2. Questo si aspetterà di trovare il file chiamato SCREEN DUMPER sul disco nel drive 8. Quando BSD non riesce a trovare il file, dà il messaggio FILE NOT FOUND! Se la ricerca viene portata a termine con successo, il messaggio è ALL DONE! Il file Screen Dump diviene allora un accessorio da tavolo GEOS.

Per vedere Screen Dumper in azione è necessario usare GEOS. Potete ottenere uno screen dump istantaneo in qualsiasi

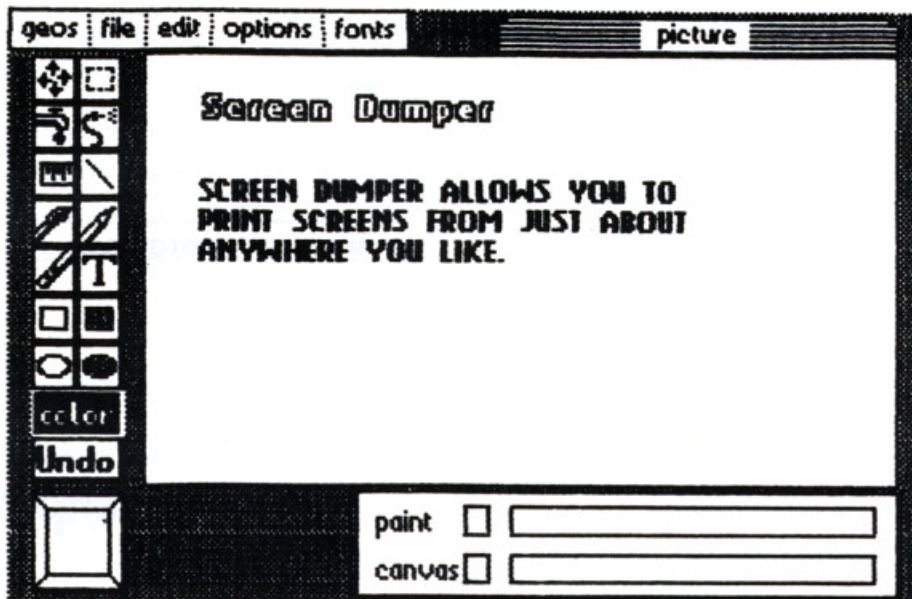
momento, basta selezionare due volte l'icona Screen Dumper o selezionare la voce Screen Dumper sul menu GEOS. Perché Screen Dumper funzioni correttamente, bisogna avere una copia del proprio printer driver sul disco inserito al presente nel drive. Se il printer driver non è memorizzato, Screen Dumper ritorna al deskTop senza stampare.

Come funziona

Usando il sistema di sviluppo GEOS PROGRAMMER, è molto facile stampare della grafica con GEOS. Se state scrivendo un'applicazione GEOS, potete stampare la grafica seguendo i 5 passi di questo procedimento.

Primo. La vostra applicazione effettua una chiamata alla routine GetDimensions del printer-driver per scoprire le dimensioni della pagina previste dalla stampante. La routine GetDimensions riporta la

Todd Crusan



lunghezza della linea nel registro x e il numero delle linee nel registro y. La lunghezza delle linee ed il numero di linee per pagina sono ora in unità della Commodore bitmap card ad alta risoluzione (una card è equivalente nelle dimensioni ad un carattere su uno schermo di testo del C64). La massima lunghezza delle linee prevista da GEOS è di 80 card, ma alcune stampanti prevedono un massimo di sole 60 card. Uno schermo GEOS è largo 40 card ed alto 25.

Secondo. La vostra applicazione deve chiamare InitForPrint. Questa routine inizializza la stampante per una sessione di stampa. Questa chiamata viene effettuata solo una volta per ogni sessione di stampa. La routine InitForPrint non riporta alcun valore.

Terzo. Chiamate StartPrint. Questa chiamata inizializza la stampante per stampare una pagina di grafica. Questa chiamata deve venir effettuata una volta per pagina. La routine StartPrint non ricerca parametri. Essa riporta semplicemente un valore di buon esito nel registro x. Un valore diverso da 0 indica un errore della stampante.

Quarto. La vostra applicazione richiama PrintBuffer per ogni linea di grafica nella pagina. L'applicazione deve prima caricare un buffer da 640 byte (80 card) con i dati della grafica e passare l'indirizzo di questo buffer in R0 (locazioni di memoria 2 e 3). L'applicazione deve anche fornire un altro buffer da 640 byte che le routine del printer driver useranno come area di lavoro. L'indirizzo di questo buffer di lavoro viene passato in R1 (locazioni di memoria 4 e 5). Le applicazioni devono occuparsi di passare lo stesso buffer di lavoro a PrintBuffer per ogni linea della pagina. La routine PrintBuffer non riporta alcun valore.

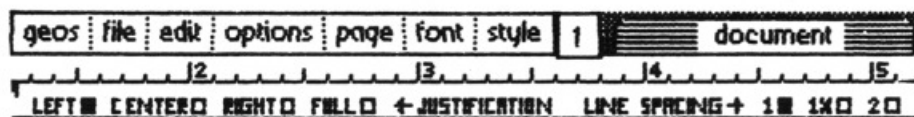
Quinto. Infine, come quinto ed ultimo passo, la vostra applicazione deve chiamare StopPrint una volta per ogni pagina per terminare la stampa. Questa chiamata azzerà il buffer della stampante e manda un form feed alla stampante.

Dato che StopPrint stampa i dati della grafica rimanenti nel buffer, l'applicazione deve mandare gli indirizzi del buffer dati e del buffer di lavoro in R0 e R1 nello stesso modo in cui li passa per la PrintBuffer. La routine StopPrint non riporta alcun valore.

Punti d'entrata della routine del Printer Driver

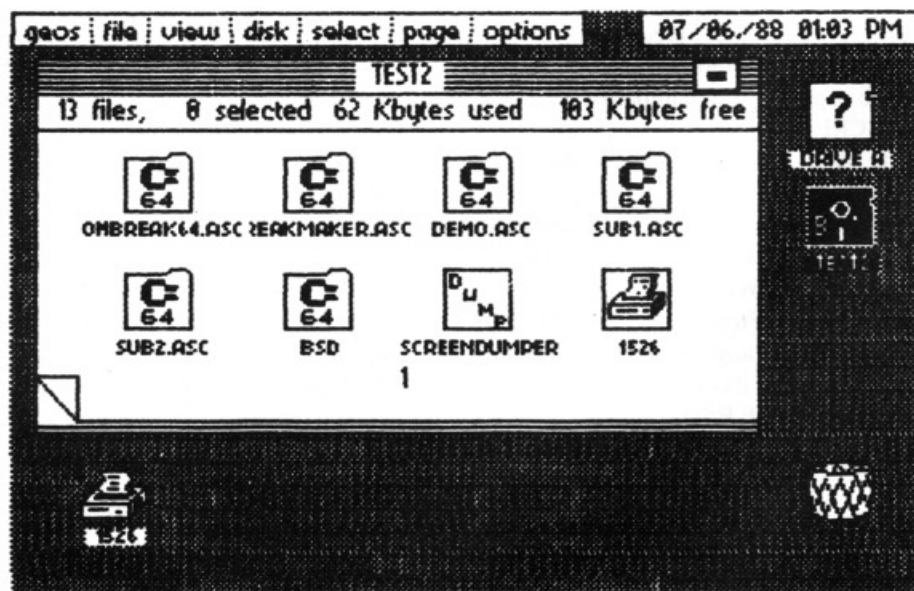
InitForPrint \$7900
(jump table + 0)

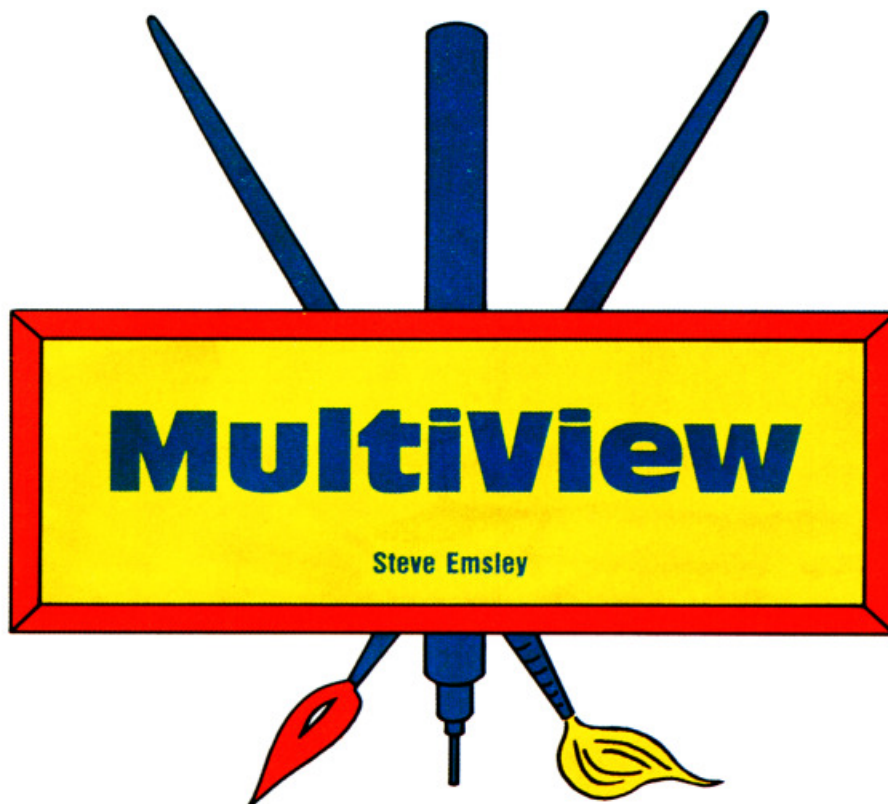
StartPrint \$7903
(jump table + 3)
PrintBuffer \$7906
(jump table + 6)
StopPrint \$7909
(jump table + 9)
GetDimensions \$790C
(jump table + 12)
PrintASCII \$790F
(jump table + 15)
StartASCII \$7912
(jump table + 18)
SetNLQ \$7915
(jump table + 21)



Since *Screen Dumper* is installed as a Desk Accessory, it's simple to use.

Just Point and Click.





MutiView consiste in due programmi, il primo scritto interamente in Basic ed è il programma principale; mentre il secondo è in linguaggio macchina.

Per introdurre il secondo programma bisogna utilizzare il programma MLX, alle domande di MLX bisogna rispondere come segue:

```
INDIRIZZO INIZIALE: C000
INDIRIZZO FINALE : C23F
```

Assicuratevi di aver salvato una copia prima di abbandonare MLX, questo secondo programma si deve chiamare MULTIVIEW.ML poiché il primo programma cercherà un file con questo nome.

Quando siete pronti per utilizzare Multiview, caricate il programma 1 e digitare RUN, il secondo programma si caricherà automaticamente.

Appena caricato il programma, vi verrà mostrato il menu principale.

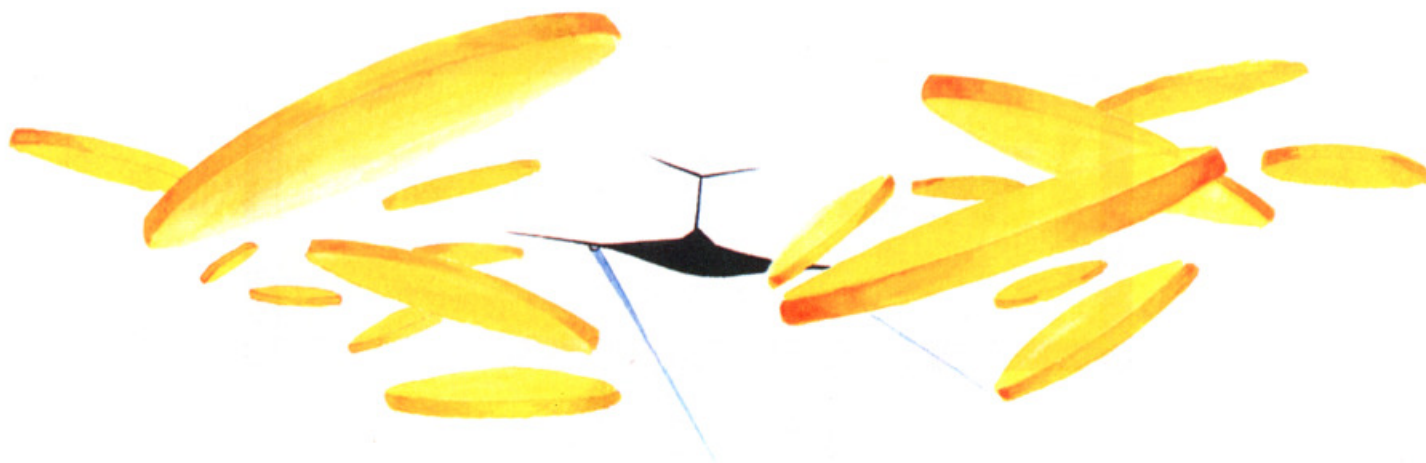
Premete 1 per caricare e vedere un'immagine. Vedrete il menu relativo al formato dell'immagine. Potrete scegliere tra questi formati: Doodle, Koala Pad, Create with Garfield, Print Shop, Paint now!, Advanced Art Studio, Rainbow Painter, Paint Brush, Blazing Paddles o il formato standard delle immagini a bassa risoluzione.

Usate i tasti cursore per selezionare la scelta e premete RETURN. Poi battete il nome del figura che intendete caricare. Assicuratevi di avere scritto gli eventuali prefissi o suffissi speciali che appaiono nel nome del file. Per esempio, i file Koala Pad sono preceduti dal carattere della vanga rovesciata. Dovete battere "(virgolette), Commodore-1, il nome del file, e " per caricare un file Koala Pad. Inoltre

**I paint program
posseggono una varietà
di formati di memorizzazione differenti, e talvolta
ciò può rappresentare
una limitazione per l'utente. Ecco un programma
compatibile con immagini create da 10 tra i più
popolari paint program.
Indispensabile il disk drive.
Non esiste uno standard per la grafica del
Commodore 64. Infatti, quasi ogni singolo paint
program utilizza il proprio metodo
per la memorizzazione delle immagini.
Fino a quando ci si limita
ad usare un solo paint program va tutto bene.
Ma cosa succede se vogliamo vedere un'immagine
downloaded da un bulletin board o da un information service?
Multiuview è la risposta.
Con Multiview potrete iniziare la vostra
collezione di opere d'arte.
Sono compatibili 10 popolari formati
inclusi quelli di *Doodle*, *Koala Pad* e *Blazing Paddles*.**

ricordatevi di battere gli spazi al termine del nome del file.

Multiview mostra l'immagine. Quando questa non vi interessa più premete f1 per tornare al menu principale. Per vedere la directory di un disco, selezionate l'opzione 2 del menu principale. Premete 3 per abbandonare il programma e tornare al BASIC. Il programma, come pure tutti gli altri, è registrato sul supporto magnetico.



Disc Blitz

Credete nei dischi volanti? Se state per rispondere di no, vi ricrederete presto. Disc-Blitz vi mette di fronte ad infinite orde di dischi volanti alieni. Avete solo poche vite e molto brevi, usatele al meglio!!

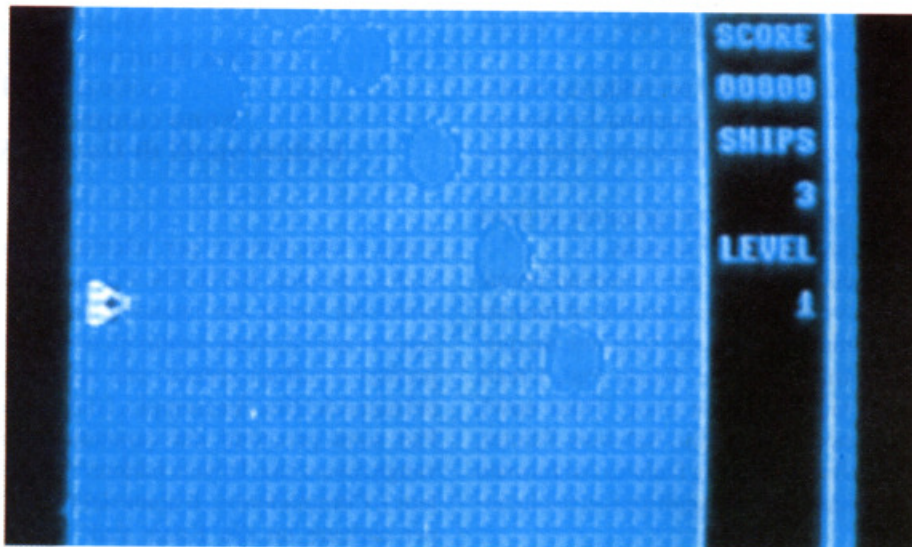
Inserimento del programma

Disc-Blitz è scritto interamente in BASIC, ma contiene anche dei codici in linguaggio macchina nelle istruzioni DATA. Inserite il programma e ricordatevi di salvare una copia del programma su nastro o disco prima di farlo funzionare. Quando vi sentite pronti ad affrontare i nemici, caricate Disc-Blitz e digitate RUN, a questo punto vedrete il titolo che introduce il gioco e per iniziare premete il pulsante di fuoco per iniziare. In Disc-Blitz siete i piloti di un caccia che svolazza alla sinistra dello schermo, potete muovervi solo su e giù. Per sparare premete il pulsante di fuoco, tenetelo sempre premuto per avere l'effetto del fuoco rapido. I dischi volanti si presenteranno in parecchie formazioni diverse, dovete imparare a vostre spese come sopravvivere ai loro pericolosi attacchi.

Dovete fronteggiare formazioni sempre diverse di inesorabili alieni. Musica spaziale e sfondo che "scrolla" creano l'ambiente di questo programma per il C64 e C128 in modo 64.

Peter Marino

Ogni volta che un disco volante vi viene addosso, perdete una vita. All'inizio del gioco ne possedete quattro, quindi dovete giocare con cautela. Ci sono cinque livelli, ma sarà necessaria un pò di pratica prima di riuscire a superarli tutti. Il programma, essendo scritto in BASIC può essere facilmente modificato; potete quindi cambiare i livelli, aggiungere dei bonus, o aumentare il numero delle vite.



bombardment

Speravate di non dover arrivare a questi estremi. Voi ed un vostro amico avete cominciato a discutere nel palazzo del Consiglio Galattico. Ma giustamente, è severamente proibito discutere nelle aule del Consiglio, altrimenti come si potrebbe mandare avanti la Via Lattea. Così siete stati condannati a regolare il vostro disaccordo nel Pozzo del Plasma. Solo uno di voi due riemergerà. Il vincitore avrà almeno imparato la lezione: non vale la pena di discutere.

Così eccovi qua, voi ed il vostro amico, ora mortali nemici. Uno di fronte all'altro, nel pozzo, PlasmaBlaster contro PlasmaBlaster. E' triste pensare che uno di voi due morirà solo perché non riuscite a mettervi d'accordo su chi di voi due dovesse pagare il caffè.

Per iniziare

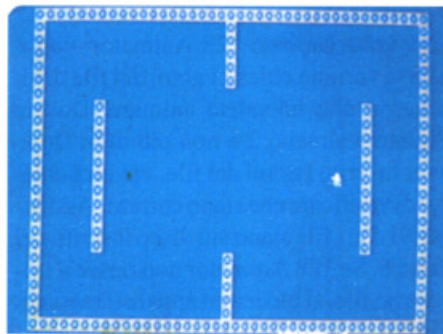
Bombardment è interamente scritto in linguaggio macchina. Inserirlo usando "MLX", il programma d'inserimento in linguaggio macchina che trovate in questo stesso numero. Quando MLX attenderà vostre istruzioni, inserite i seguenti valori:

INDIRIZZO INIZIALE: C000
INDIRIZZO FINALE: C90F

Digitate i dati del programma. Prima di uscire da MLX, assicuratevi di aver salvato una copia dei dati su nastro o disco.

**Un frenetico gioco
d'azione per il 64.
Due giocatori
si confrontano
con un'arma mortale:
il PlasmaBlaster.
Attenzione, basta
una piccola disattenzione
e si corre il rischio
di autodistruggersi.
Sono necessari due joystick.**

Forrest Bentley



Per dare il via al gioco, inserite due joystick nel computer. Ora caricate il programma con la solita formula:

LOAD"nome del file",8,1
Gli utenti che utilizzano il nastro digiteranno:

LOAD"nome del file",1,1
Specificate il nome che avete dato al programma quando lo avete salvato con MLX. Battete NEW e poi SYS 49152. Vi troverete di fronte al Pozzo del Plasma.

Schivare il Plasma

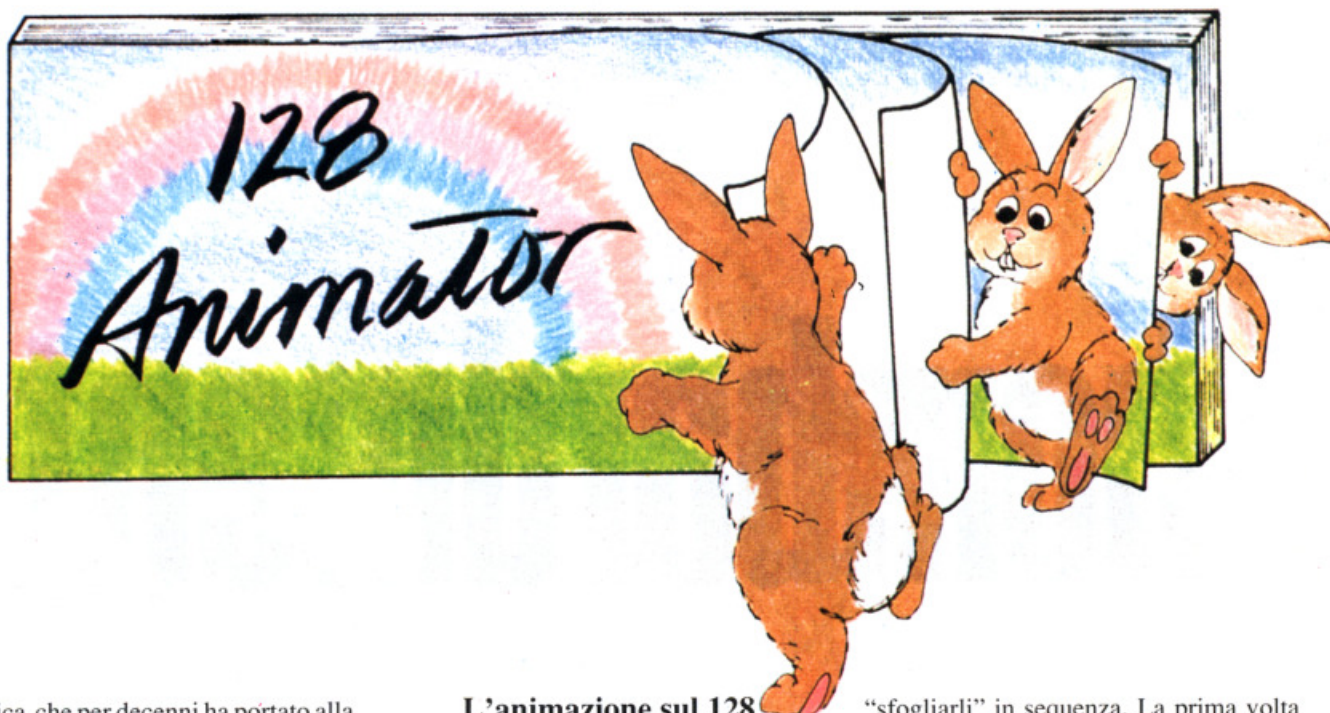
Il joystick inserito nella porta 1 muove il giocatore sul lato destro dello schermo. L'altro joystick controlla il giocatore a sinistra.

Muovete il vostro giocatore in tutte le direzioni con il joystick. Usate il vostro mitra al plasma premendo il pulsante di fuoco. Ogni volta che premete il grilletto, sparate una mortale palla di plasma. Il pozzo è magneticamente sigillato, così il plasma rimbalza per la stanza fino a quando non colpisce qualcosa (per fare dei nomi, voi o il vostro avversario).

Maggiore è il numero delle palle di plasma che voi ed il vostro antagonista sparate, e più il gioco diventa frenetico.

Se sparate mentre siete in corsa, la palla di plasma sarà ancora più veloce delle altre. Attenzione a non farvi colpire dai vostri stessi colpi, sono mortali quanto quelli dell'avversario.

Entrambi i giocatori incominciano a giocare con cinque vite. Ovviamente, il vincitore è colui che riesce a sopravvivere. Alla fine di ogni partita, vi verrà chiesto se intendete giocare ancora: premete Y per un'altra partita o N per abbandonare e tornare al BASIC.



La tecnica che per decenni ha portato alla vita i personaggi dei cartoni è quella delle pagine che si sfogliano velocemente. Tutti noi la possiamo sperimentare, basta prendere una matita ed un blocco di fogli. Disegniamo ora una figura stilizzata su un foglio, poi la ridisegniamo sul successivo cambiandone leggermente la posizione, così sul terzo, sul quarto e via. Ora si afferra il blocco da un lato e ne si fanno scorrere velocemente le pagine. Ed ecco l'animazione! "128 Animator" è un programma per il 128 che consente il caricamento di sei schermate ad alta risoluzione in una volta sola. Una volta che sono in memoria, "128 Animator" può sfogliare gli schermi, proprio come facevamo prima noi con i fogli, e creare l'illusione del movimento. E' possibile anche fermarsi su un "fotogramma", per esaminarlo attentamente. Disegnate le vostre schermate con il programma di grafica che preferite e poi sarà 128 Animator a metterle in moto. E' compatibile con gli schermi disegnati con la maggior parte dei programmi per disegnare per il C64 e per il C128. Ma se preferite, potete disegnare con il generatore di schermi incorporato.

Inserimento del programma

Il programma 1, "128 Animator" è scritto in BASIC. Quando avrete inserito il programma 1 nel computer salvatene una copia su disco con il nome 128ANIM.

L'animazione sul 128 è uno scherzo con questo programma di grafica ad alta risoluzione. Avete solo bisogno di un disk drive.

Earl Kanaday

Anche il programma 2, "Screen Generator" è scritto in BASIC. Inserirlo e salvatene una copia su disco.

I primi passi

Una volta caricato 128 Animator con il 128 vi verranno chiesti i nomi dei file degli schermi che intendete animare. Dovete digitarne almeno 2 e non più di 6. Dopo aver inserito i nomi dei file, vi verrà chiesto di verificare che siano corretti. Assicuratevi che i file siano sul disco inserito nel drive 8. Se 128 Animator non riesce a trovare un file, si blocca ed appare il messaggio FILE NOT FOUND ERROR.

Una volta caricati i file, compare il menu principale di 128 Animator. Potete selezionare varie opzioni: visionare la sequenza automaticamente, vederla manualmente, scegliere una nuova sequenza di file, o abbandonare il programma.

Se decidete di vedere gli schermi automaticamente, 128 Animator comincia a

"sfogliarli" in sequenza. La prima volta che userete il programma, gli schermi si susseguiranno alla massima velocità possibile. Decelerate il ritmo premendo il tasto - (meno) o SHIFT -. Premete + o SHIFT + per aumentare la velocità. RETURN ritorna al menu principale.

Nel caso della visione manuale, il programma chiederà il numero corrispondente allo schermo che volete guardare. Premete il tasto che riguarda lo schermo che vi interessa e, se a questo punto volete vederne un altro, premete un altro tasto. RETURN per il menu principale.

Come creare sequenze di animazione

128 Animator è compatibile con qualsiasi file di grafica ad alta risoluzione realizzato sul C64 o C128. L'unica restrizione che il programma impone, è che i data del colore e i data bitmap devono essere salvati insieme. *Doodle!* è uno di quei programmi che salvano insieme i data bitmap e quelli relativi al colore (se il file della figura occupa 36 o 37 blocchi sul disco, significa che colore e immagine sono stati salvati insieme).

Se i data bitmap e del colore sono stati invece salvati come file separati, è possibile unirli caricandoli separatamente per poi salvarli come un unico file. Ciò può venire eseguito seguendo queste istruzioni:


```

BLOAD "nome del file
      relativo al
      colore",P7168
BLOAD "nome del file
      relativo al
      bitmap",P8192
BSAVE "nuovo nome del file",
      P7168 TO P16384

```

Gli schermi a colori di *FlexiDraw* vengono salvati con questo formato.

Per caricare gli schermi monocromatici ad alta risoluzione (per esempio come quelli di *FlexiDraw* o *Computer Eyes* senza colore), dovete decidere il colore 0 e il colore 1, caricare il bitmap, e poi risaltare colore e bitmap come unico file:

```

COLOR 0,1:COLOR1,2
BLOAD "nome del file
      relativo al
      bitmap",P8192
BSAVE "nuovo nome del
      file",P7168 TO P16384

```

Disegno veloce

Potete usare il programma 2, *Screen Generator*, per disegnare semplici sequenze di animazione. Questo realizza sei schermi di grafica e li salva su disco con i nomi PIC1-PIC6. Prima di caricare il programma 2 assicuratevi che il disco inserito nel drive 8 abbia almeno 222 blocchi liberi. Apportando cambiamenti minimi al programma 2, potete creare sequenze animate diverse. Provate a fare queste modifiche:

```

60   C$=RIGHT$(STR$(C),1)
      :BSAVE"PICA"+C$,P7168
      TO P16384
70   L=L-101:H=H-60
      :C=C+1:IFC<=-6 THEN40

```

Oppure queste:

```

30   X=159:Y=99:L=86:H=50:
      S=0:E=360:N=0:I=60:C=1
60   C$=RIGHT$(STR$(C),1)
      :BASVE"PICB"+C$,P7168
      TO P16384

```

```

70   L=L-101:H=H-70
      :C=C+1:IFC<=6THEN40

```

Come funziona

Il chip VIC-II indirizza quattro banchi video nei banchi RAM del 128 per un totale di 8 banchi video e poiché i registri dell'MMU occupano la memoria sopra i 65280, usare il banco video 3 non è molto pratico. I sei banchi video sono disponibili POKando in 2 locazioni di memoria: la 54534 e la 56576. La locazione 54534, il registro della configurazione RAM, commuta il 128 tra la RAM 0 e la RAM 1. La locazione 56576, la porta A di I/O del registro data, commuta tra i 4 banchi video in ognuno dei banchi RAM. POKando alcuni valori specifici all'interno di queste due locazioni, è possibile spostarsi tra le sei aree degli schermi. 128 Animator carica i sei schermi hi-res, uno per ognuna delle banche video accessibili. Con i file in memoria, per farli apparire fare il POKE dei valori nei due registri.

(segue da pagina 25)

I.m.

visualizziamo e lo depositiamo nella stringa (posizione Y, giusto?). Poi aggiungiamo 1 a Y, lo immagazziniamo in \$8E, e torniamo indietro:

```

JSR   $FFD2 (Stampa)
STA   ($8C),Y
      (Deposita nella
      stringa)
INY   (Incrementa
      conto)
STY   $8E
BNE   $038C (Sempre
      branch)

```

Quando vediamo un RETURN, abbiamo finito, dobbiamo solo visualizzarlo (muovendoci ad una nuova linea di schermo) e poi dobbiamo tornare al BASIC. Potremmo codificare JSR (Jump to SubRoutine) seguito da RTS (ReTurn from

SubRoutine), ma è facile quanto JMP (JuMP) alla subroutine e lasciare che ritorni al BASIC a nostro vantaggio:

```

03A5 JMP   $FFD 2
      (Stampa ed esce)

```

Per un esempio in BASIC guardate il programma.

Vedere listato a pagina 60.

(segue da pagina 24)

Divagazioni

scoprire metodi ingenui per esplorare materie e soggetti critici del programma, usando appunto tali tecnologie. I docenti possono svolgere il ruolo di guida collaborando con i propri studenti, come in una squadra di esploratori della conoscenza che traduce la materia dei libri in nuove, eccitanti rappresentazioni multimediali.

Se gli insegnanti intendono preparare gli studenti a questo luogo di lavoro e di studio degli anni '90, debbono creare un ambiente che si avvicini il più possibile a quel luogo di lavoro e, in questo senso, la tecnologia non è altro che uno degli ingredienti che costituiscono tale ambiente. I professori debbono anche considerare l'opportunità che gli studenti si assumano la responsabilità del loro apprendimento e diventino "produttori" della loro stessa conoscenza. Gli studenti, da parte loro, devono avere il tempo di lavorare in gruppi, come team di collaboratori, ed essere messi nella situazione di poter esplorare e pensare in maniera critica per risolvere problemi e portare a termine compiti. Tali classi sarebbero veramente classi multimediali, modelli per l'ambiente di lavoro degli anni '90, nelle quali gli insegnanti svolgerebbero il ruolo di guida ed il lavoro procederebbe in stretta collaborazione con gli studenti.

Listato di: I problemi della programmazione

Articolo a pagina 16

```

10  REM INIZIALIZZA
    VARIABILI
20  CO=54272:RD=5:REM
    COLORE MEMORIA,
    NUMERI DI ROUND
30  S1=0:S2=0:REM SCORE
    DEI GIOCATORI
40  B=102:BC=4:REM COLORE
    E CARATTERI DEL BORDO
50  R1=81:C1=2:R2=42
    :C2=6:REM COLORI E
    CARATTERI DEI GIOCATORI
60  B$="PRESS A FIRE
    BUTTON TO PLAY"
70  C$="{36 SPC}"
80  REM
90  REM INIZIALIZZA
    VARIABILI
100 P1=1474:P2=1492:REM
    POSIZIONE INIZIALE
    DEI GIOCATORI
110 P3=1:P4=-1:REM
    INCREMENTO INIZIALE
    DEI GIOCATORI
120 SC=0:REM SCORE
130 REM
140 PRINT "{CLR}":REM
    SETUP CAMPO DI GIOCO
150 POKE 53281,15:POKE
    53280,15:REM COLORI
    SCHERMO
160 REM DISEGNA BORDO
170 FOR I=1064 TO 1064+
    39:POKE I+CO,BC:POKE
    I,B:NEXT:REM ALTO
180 FOR I=1944 TO 1944+
    39:POKE I+CO,BC:POKE
    I,B:NEXT:REM BASSO
190 FOR I=1104 TO 1904
    STEP 40:POKE I+CO,
    BC:POKE I,B:NEXT:REM
    LATO SINISTRO
200 FOR I=1143 TO 1943
    STEP 40:POKE I+CO,
    BC:POKE I,B:NEXT:REM
    LATO DESTRO
210 REM VISUALIZZA IL
    PUNTEGGIO DEL
    GIOCATORE
220 PRINT "{HOME}{RED}{2
    SPC}PLAYER 1{14SPC}
    SCORE:"S1
230 PRINT "{23 CUR.GIU}
    {BLUE}{2 SPC}PLAYER
    2{14 SPC}SCORE:"S2;
240 IF RD=0 THEN 560:REM
    FINE GIOCO
250 POKE CO+P1,C1:POKE
    P1,R1:POKE CO+P2,C2:
    POKE P2,R2:REM
    POSIZIONE GLOCATORE
260 PRINT "{HOME}{4 CUR.
    GIU}"SPC(2)"ROUND"5-
    RD+1;"{CUR.SIN}:" ";B$
270 GOSUB 640:REM
    CONTROLLA FIRE DEL
    JOYSTICK
280 PRINT "{HOME}{4 CUR.
    GIU}"SPC(2)C$
290 REM
300 REM LOOP GIOCO
310 J1=15-(PEEK(56321)
    AND15):IF J1=0 THEN
    360
320 IF J1=1 THEN P3=-40:
    GOTO 360:REM #1
    JOYSTICK IN ALTO
330 IF J1=2 THEN P3=40:
    GOTO 360:REM #1
    JOYSTICK IN BASSO
340 IF J1=4 THEN P3=-1:
    GOTO 360:REM #1
    JOYSTICK A SINISTRA
350 IF J1=8 THEN P3=1:REM
    #1 JOYSTICK A DESTRA
360 P1=P1+P3:REM UPDATE
    POSIZIONE GIOCATORE 1
370 IF PEEK(P1)<>32 THEN
    W=2:GOTO 510:REM #1
    COLLISIONE?
380 POKE P1+CO,C1:POKE
    P1,R1:REM NO, COSI'
    MUOVE #1
390 J2=15-(PEEK(56320)
    AND15):IF J2=0 THEN
    440
400 IF J2=1 THEN P4=-40:
    GOTO 440:REM #2 ALTO
410 IF J2=2 THEN P4=40:
    GOTO 440:REM #2 BASSO
420 IF J2=4 THEN P4=-1:
    GOTO 440:REM #2
    SINISTRA
430 IF J2=8 THEN P4=1:REM
    #2 DESTRA
440 P2=P2+P4:REM UPDATE
    POSIZIONE GIOCATORE 2
450 IF PEEK(P2)<>32 THEN
    W=1:GOTO 510:REM #2
    COLLISIONE?
460 POKE P2+CO,C2:POKE
    P2,R2:REM NO, COSI'
    MUOVE #2
480 GOTO 310:REM PROSSIMA
    MOSSA
490 REM
500 REM COLLISIONE
    AGGIORNA SCORE
510 IF W=1 THEN S1=
    S1+SC*10+10:GOTO
    530:REM #1 SCORE
520 S2=S2+SC*10+10:REM
    #2 SCORE
530 FOR I=1 TO 750:NEXT
    :RD=RD-1:GOTO 100:
    REM ATTESA PROSSIMO
    ROUND
540 REM
550 REM GAME OVER

```

```

560 PRINT "{HOME}{10 CUR.
    GIU}"SPC(15)"GAME
    OVER"
570 IF S1=S2 THEN PRINT
    SPC(14)"IT'S A
    TIE!":GOTO 600
580 W=1:IF S2>S1 THEN
    W=2
590 PRINTSPC(13)"PLAYER
    "W"WINS!"
600 PRINTSPC(3)"{CUR.
    GIU}"B$" AGAIN"
610 GOSUB 640:RUN:REM
    PROSSIMA PARTITA
620 REM
630 REM ROUTINE FIRE
640 J1=PEEK(56321) AND
    16:J2=PEEK(56320)AND
    16:IF J1=16 AND
    J2=16 THEN 640
650 RETURN

```

Listato di: Programmazione in I.m.

Articolo a pagina 25

```

100 REM FUNZIONA SOLO
    SU C64
110 REM LE STRINGHE
    DEVONO ESSERE
    DICHIARATE SEMPRE
    PRIMA
120 REM L'ESEMPIO SOTTO
    PERMETTE 5 CARATTERI
130 A$="{5 SPC}"+" "
140 DATA 160,2,177,45,
    153,137,0,200
150 DATA 192,6,208,246,
    32,228,255
160 DATA 240,251,201,13,
    240,16,164,142
170 DATA 196,139,176,241,
    32,210,255
180 DATA 145,140,200,132,
    142,208,231
190 DATA 76,210,255
200 FOR J=896 TO 935
210 READ X
220 T=T+X
230 POKE J,X
240 NEXT J
250 IF T<>6296 THEN STOP
260 PRINT"DIGITA LA
    STRINGA: ";
270 SYS 896
280 L=PEEK(142)
290 PRINT"RICEVO: ";LEFT$
    (A$,L)
300 PRINT"DIGITA LA
    STRINGA: ";
310 SYS 896
320 L=PEEK(142)
330 PRINT"RICEVO: ";LEFT$
    (A$,L)

```


Listato di: La potenza del BASIC

Articolo a pagina 52

Programma 1:

On Break Goto - C64

```

10 REM COPYRIGHT 1989
   SUPERCOMMODORE-
   COMPUTE!
20 PRINT "{CLR}":POKE
   53281,6:POKE53280,14
   :PRINTTAB(11)"{GRAY2}
   COPYRIGHT 1989"
30 PRINTTAB(13)
   "SUPERCOMMODORE-
   COMPUTE! "
40 PRINTTAB(9)"ALL
   RIGHTS RESERVED{3CUR.
   GIU}":CK=0
50 FORI=49152TO49270
   :READA:CK=CK+A:POKEI,A
   :NEXT
60 IF CK <> 12952 THEN
   PRINT"ERROR IN DATA
   STATEMENTS.":END
70 I=I-1:LN$="280":REM
   ON BREAK LINE
80 FOR A=1 TO LEN(LN$)
   :POKE I+A,ASC(MID$
   (LN$,A,1)):NEXT
90 POKE I+A,13:POKE I+A+
   1,19:POKE I+A+2,0
   :SYS 49152
100 PRINTTAB(10);"ON
   BREAK ACTIVATED."
110 PRINT"PRESS THE STOP
   KEY TO JUMP TO LINE "
   LN$
120 GOTO120
130 DATA 173,40,3,201,
   49,208,20,173
140 DATA 41,3,201,192,
   208,13,173,103
150 DATA 192,141,40,3,
   173,104,192,141
160 DATA 41,3,96,173,
   40,3,141,103
170 DATA 192,173,41,3,
   141,104,192,169
180 DATA 49,141,40,3,
   169,192,141,41
190 DATA 3,165,145,201,
   127,208,47,169
200 DATA 147,32,210,255,
   169,105,133,251
210 DATA 169,192,133,252,
   160,0,177,251
220 DATA 240,6,153,0,4,
   200,208,246
230 DATA 169,17,32,210,
   255,169,2,133
240 DATA 198,169,19,141,
   119,2,169,13
250 DATA 141,120,2,169,
   0,168,96,0
260 DATA 0,16,18,9,14,

```

```

20,34,211
270 DATA 34,58,7,15,20,
   15,32
280 REM
290 REM YOUR CODE GOES
   HERE...
300 REM
310 PRINT"NOW AT LINE "
   LN$:SYS 49152
320 END

```

Programma 2: Break Maker - C128

```

10 REM COPYRIGHT 1989
   SUPERCOMMODORE -
   COMPUTE!
15 POKE53280,13:POKE
   53281,11
20 PRINT "{CLR}{LT.BLUE}{12
   SPC}COPYRIGHT 1989":
   PRINT "{7SPC}COMPUTE!"
30 PRINT "{2 CUR.GIU}
   PLEASE MAKE CERTAIN
   THAT YOU HAVE{7 SPC}
   INCLUDED YOUR MESSAGE
   AS A$ IN LINE 50":
   PRINT:PRINT
40 AD=2816:PRINT"
   INDIRIZZO DI PARTENZA
   (DECIMALE) ";AD
50 PRINT "{CUR.SU}";TAB
   (27):INPUT AD
60 A$="RUN"+CHR$(34)+
   "SUB1"+CHR$(34)
70 IFLEN(A$)>95THEN
   PRINT"STRING TOO LONG
   ERROR: BRANCH
   OVERFLOW.":STOP
80 FORX=ADTOAD+24:READ
   A:POKEX,A:Y=Y+A
   :NEXT:IFY<>2972THEN
   PRINT "{CUR.GIU}ERROR IN
   DATA STATEMENTS."
   :STOP
90 FORX=1TOLEN(A$):A=
   ASC(MID$(A$,X,1)):POKE
   AD+24+X,A:NEXT
100 FORX=AD+25+LEN(A$)
   TOAD+25+LEN(A$)+16:
   READA:POKEX,A:NEXT
110 POKEAD+10,30+LEN(A$)
120 PRINTCHR$(147):PRINT
   :PRINT:PRINT"BSAVE"
   CHR$(34)"ONBREAK/
   CMD"CHR$(34)"",B15,P"AD"
   TO P"AD+42+LEN(A$)
   ;CHR$(19)
130 POKE208,1:POKE842,13
   :END
140 DATA169,0,141,0,255
   ,165,145,201,127,208,39,
   173,17,208,41,239,
   141,17,208,32,23,250,147,
   13,13
150 DATA19,0,169,1,133,
   208,169,13,141,74,3,
   104,104,76,3,64,96

```

Programma 3: Demo - C128

```

0 PRINTCHR$(147)"THIS
   IS A DEMO PROGRAM. {2
   SPC}ITLOADS{7SPC}
   ONBREAK AND SETS UP
   POINTERS FOR IT."
5 BANK 15:BLOAD"ON
   BREAK/CMD":POKE 808,
   0:POKE809,11:POKE792,98
10 PRINT"PRESS RETURN TO
   CONTINUE.":GETKEYA$
15 RUN"SUB1"

```

Programma 4: Demo Sub 1 - C128

```

0 PRINTCHR$(147);:SLOW
5 PRINT"THIS IS SUB
   PROGRAM 1"
10 PRINT "{CUR.GIU}PRESS
   RETURN TO EXECUTE SUB
   PROGRAM {SPACE}2."
15 GETKEYA$
20 RUN"SUB2"

```

Programma 5: Demo Sub 2 - C128

```

0 PRINTCHR$(147)"THIS
   IS SUB PROGRAM 2"
5 PRINT "{CUR.GIU}PRESS
   <RUN STOP> KEY TO
   RETURN TO SUB{3 SPC}
   PROGRAM 1."
10 GOTO 10

```

**Risposte al quiz
Conosci l'Informatica
e il tuo C64/128**

- | | |
|-----|---|
| 1) | c |
| 2) | c |
| 3) | a |
| 4) | b |
| 5) | b |
| 6) | b |
| 7) | b |
| 8) | a |
| 9) | a |
| 10) | c |

MLX II

MLX è un programma che permette di introdurre listati in linguaggio macchina senza errori, per C 64 e C 128 in modo 64.

Quando bisogna introdurre un programma, della nostra rivista scritto in linguaggio macchina, bisogna caricare e dare il RUN al programma MLX II. Gli indirizzi richiesti si possono leggere nell'articolo che accompagna il programma formato MLX.

Se siete "digiuni" di linguaggio macchina questi indirizzi (e tutti gli altri valori da introdurre) possono apparire strani. Invece questi numeri sono propri del sistema esadecimale a base 16 comunemente usato appunto per i programmi in linguaggio macchina. Il sistema esadecimale include i numeri da 0 a 9 e le lettere da A a F. Dopo aver introdotto l'indirizzo iniziale e finale viene offerta l'opzione per pulire l'area di lavoro con tutti zeri.

A questo punto apparirà sul video il menu.

Per introdurre un programma bisogna scegliere la prima opzione cioè: immettere dati.

Premendo la lettera I verrà richiesto l'indirizzo della prima linea.

Se il programma in parte è stato digitato, bisogna assicurarsi di aver caricato quella parte, scegliere la prima opzione e digitare il numero di linea della rimanente parte da introdurre. Bisogna sempre assicurarsi che l'indirizzo immesso corrisponda a quello della linea del listato, altrimenti i dati non vengono accettati.

Per tornare al menu basta premere RETURN da qualsiasi opzione.

Le linee da introdurre sono composte da nove cifre, le prime otto sono dati, l'ultima è il checksum. Se la linea è stata introdotta in modo corretto verrà emesso un beep ed il programma passerà alla linea successiva, viceversa la linea non verrà accettata. Altro vantaggio di MLX è quello di non dover introdurre il numero di linea.

Il checksum di MLX è molto potente, se si digita A0 invece di 0A questo viene notato ed apparirà il messaggio di errore. Al contrario, se si immette FF invece di 00 e viceversa, non viene notato, per cui questo è l'unico caso limite e naturalmente bisogna fare

attenzione. Per cancellare un errore di battitura si deve usare il tasto INST/DEL o il cursore verso sinistra.

Il tasto RETURN funziona solo prima di digitare qualche dato della linea.

Il tasto CLR/HOME pulisce la linea dopo aver visto il messaggio di errore. La seconda opzione del menu riguarda la visualizzazione dei dati.

Quando si preme V viene chiesto l'indirizzo iniziale: questa opzione mostra tutti i dati da tale indirizzo fino alla fine della memoria, dopodiché il menu viene visualizzato.

Per fare una pausa durante la visualizzazione dei dati bisogna premere la barra spazio, così pure per tornare alla visualizzazione.

Con RETURN si torna al menu.

Altre due opzioni sono di LOAD e di SAVE di un programma.

Naturalmente è possibile registrare o salvare su disco o su nastro.

Quando si salva un programma bisogna sempre usare nomi differenti. MLX riporta tutti i messaggi standard del Commodore 64.

ed in più include altri tre messaggi:

1) **INDIRIZZO INIZIALE ERRATO**, questo significa che il programma che si tenta di caricare non inizia all'indirizzo specificato all'inizio quando si è fatto girare MLX.

2) **CARICAMENTO FINITO** a indirizzo..., questo messaggio significa che il programma che si prova di caricare finisce prima dell'indirizzo finale introdotto quando viene dato il RUN.

3) **FINITO ALL'INDIRIZZO FINALE**, questo messaggio significa che il programma che si tenta di caricare finisce oltre all'indirizzo finale specificato all'inizio.

Se si incontra uno di questi errori bisogna uscire da MLX tornando al BASIC, con l'opzione FINE, dare il RUN e immettere i giusti indirizzi.

Se si digita un programma lungo è consigliabile salvare, caricare e verificare frequentemente. Per far partire il programma digitato su MLX bisogna far riferimento all'articolo del programma.

LISTATO DI C64 MLX II ESA

```

10 REM MLX II ESADECIMALE
100 POKE56,50:CLR:DIMIN$,I,J,
    A,B,A$,B$,A(7),N$
110 C4=48:C6=16:C7=7:Z2=2:
    Z4=254:Z5=255:Z6=256:Z7=127
120 FA=PEEK(45)+Z6*PEEK(46)
    :BS=PEEK(55)+Z6*PEEK(56):
    H$="0123456789ABCDEF"
130 R$=CHR$(13):L$="{CUR.SIN}"
    :S$="":D$=CHR$(20):Z$=CHR$
    (0):T$="{12 CUR.DES}"
140 SD=54272:FORI=SDTOSD+23:
    POKEI,0:NEXT:POKESD+24,15:
    POKE788,52
150 PRINT"{CLR}"CHR$(142)
    CHR$(8):POKE53280,15:POKE
    53281,15
160 PRINT T$ "{RED}{RVSON}
    {2 SPC}{8 CBM@}{2 SPC}
    "SPC(28)"{2SPC}{RVSOFF}{BL
    UE}MLXII{RED}{RVSON}{2
    SPC}"SPC(28)"{12SPC}{BLUE}"
170 PRINT"{3CUR.GIU}{3SPC}
    EDITOR PER IL LINGUAG
    GIOMACCHINA.{3CUR.GIU}"
180 PRINT"{BLACK}INDIRIZZO
    DI PARTENZA{GRAY1}":
    :GOSUB300:SA=AD:GOSUB1040
    :IFFTHEN180
190 PRINT"{BLACK}INDIRIZZO
    FINALE{GRAY1}":;GOSUB300:
    EA=AD:GOSUB1030:IF F
    THEN190
200 INPUT"{3CUR.GIU}{BLACK}
    PULISCO L'AREA DILAVORO
    [S/N]{GRAY1}":A$:IF
    LEFT$(A$,1)<>"S"THEN220
210 PRINT"{2CUR.GIU}{BLUE}
    ATTENDERE...":FORI=BS
    TO BS+EA-SA+7:POKEI,0:
    NEXT:PRINT"FATTO!"
220 PRINTTAB(10)"{2CUR.GIU}
    {BLACK}{RVSON} MENU
    COMANDI MLX {CUR.GIU}
    {GRAY1}":PRINT T$"{RVS
    ON}I{RVSOFF}MMETTIDATI"
230 PRINT T$"{RVS ON}V{RVS
    OFF}ISUALIZZA DATI":
    PRINT T$"{RVS ON}L
    {RVS OFF}OAD FILE"
240 PRINT T$"{RVS ON}S{RVS
    OFF}AVE FILE":PRINT
    T$"{RVSON}F{RVS OFF}
    INE{CUR.GIU}{BLACK}"
250 GETA$:IFA$=N$ THEN250
260 A=0:FORI=1TO5:IFA$=MID$
    ("IVLSF",I,1)THENA=I:I=5
270 NEXT:ON AGOTO700,280
    :GOSUB1060:GOTO250
280 PRINT"{RVS ON} FINE
    ":INPUT"{CUR.GIU}{GRAY1}SEI
    SICURO ?[S/N}":A$:IF
    LEFT$(A$,1)<>"S"THEN220
290 POKESD+24,0:END
300 IN$=N$:AD=0:INPUTIN$:
    IFLEN(IN$)<>4THENRETURN
310 B$=IN$:GOSUB320:AD=A:

```



```

B$=MID$(IN$,3):GOSUB320:AD=
AD*256+A:RETURN
320 A=0:FORJ=1TO2:A$=MID$(
(B$,J,1):B=ASC(A$)-C4+
(A$>"@")*C7:A=A*C6+B
330 IF B<0 OR B>15THENAD=0
:A=-1:J=2
340 NEXT:RETURN
350 B=INT(A/C6):PRINTMID$(
(H$,B+1,1):B=A-B*C6
:PRINTMID$(H$,B+1,1):
:RETURN
360 A=INT(AD/Z6):GOSUB350
:A=AD-A*Z6:GOSUB350
:PRINT":":
370 CK=INT(AD/Z6):CK=AD-
Z4*CK+Z5*(CK>Z7):GOTO390
380 CK=CK*Z2+Z5*(CK>Z7)+A
390 CK=CK+Z5*(CK>Z5):RETURN
400 PRINT"{CUR.GIU}INIZIA
A{GRAY1}":GOSUB300:IF
IN$<>N$ THEN GOSUB1030
:IF FTHEN400
410 RETURN
420 PRINT"{RVS ON}IMMETTI
DATI":GOSUB400:IF
IN$=N$ THEN220
430 OPEN3,3:PRINT
440 POKE198,0:GOSUB360:IFF
THEN PRINT IN$:PRINT"
{CUR.SU}{5 CUR.DES}":
450 FORI=0TO24STEP3:B$=S$:
FORJ=1TO2:IF FTHENB$
=MID$(IN$,I+J,1)
460 PRINT"{RVS ON}"B$LS$:
IF I<24 THEN
PRINT"{RVS OFF}":
470 GETA$:IFA$=N$ THEN470
480 IF (A$>"/"ANDAS<"")OR
(A$>"@"ANDAS<"G")
THEN540
490 IFA$=R$ AND ((I=0)AND
(J=1)OR F)THENPRINTB$:
:J=2:NEXT:I=24:GOTO550
500 IFA$="{HOME}"THEN
PRINTB$:J=2:NEXT:I=24
:NEXT:F=0:GOTO440
510 IF (A$="{CUR.DES}")ANDF
THENPRINTB$LS$:GOTO540
520 IFA$<>L$ AND AS<>D$ OR
((I=0)AND(J=1))THEN
GOSUB1060:GOTO470
530 AS=L$+S$+L$:PRINT
B$LS$:J=2-J:IF J THEN
PRINTL$:I=I-3
540 PRINTA$:NEXTJ:PRINTS$:
550 NEXTI:PRINT:PRINT"
{CUR.SU}{5 CUR.DES}":
:INPUT#3,IN$:IFIN$=N$
THEN CLOSE3:GOTO220
560 FORI=1TO25STEP3:
B$=MID$(IN$,I):GOSUB
320:IFI<25THEN
GOSUB380:A(I/3)=A
570 NEXT:IF A<>CK THEN
GOSUB1060:PRINT"{BLACK}
{RVS ON}ERRORE
RIIMMETTI LA LINEA
{GRAY1}":F=1:GOTO440
580 GOSUB1080:B=BS+AD-SA
:FOR I=0TO7:POKEB+

```

```

I,A(I):NEXT
590 AD=AD+8:IFAD>EA THEN
CLOSE3:PRINT"{CUR.GIU}
{BLUE}** FINE IMMIS
SIONE **{BLACK}{2
CUR.GIU}":GOTO700
600 F=0:GOTO440
610 PRINT"{CLR}{CUR.GIU}
{RVS ON}VISUALIZZA
DATI " :GOSUB400:IF
IN$=N$THEN220
620 PRINT"{CUR.GIU}{BLUE}
PREMI:{RVS ON}SPACE
{RVS OFF}{2 SPC}PAUSA,
{2 SPC}{RVS ON}RETURN
{RVS OFF}
{2SPC}STOP{GRAY1}{CUR.GIU}"
630 GOSUB360:B=BS+AD-SA
:FORI=BT0B+7:A=PEEK(I)
:GOSUB350
:GOSUB380:PRINT S$:
640 NEXT:PRINT"{RVSON}":A=CK
:GOSUB350:PRINT
F=1:AD=AD+8:IFAD>EATHEN
PRINT"{CUR.GIU}{BLUE}**
FINE DATI **":GOTO220
660 GETA$:IFA$=R$THEN
GOSUB1080:GOTO220
670 IFA$=S$THENF=F+1:GOSUB1080
680 ONFGOTO630,660,630
690 PRINT"{CUR.GIU}{RVSON}
LOAD DATI":OP=1
:GOTO710
700 PRINT"{CUR.GIU}{RVSON}
SAVE FILE ":OP=0
710 IN$=N$:INPUT"{CUR.GIU}
NOME FILE{GRAY1}":IN$
:IF IN$=N$ THEN220
720 F=0:PRINT"{CUR.GIU}{BLACK}
{RVS ON}N{RVSOFF}ASTRO
O {RVS ON}D{RVS OFF}IS
CO: {GRAY1}":
730 GETA$:IFA$="N"THEN
PRINT"t{CUR.GIU}":GOTO880
740 IFA$<>"D"THEN730
750 PRINT"d{CUR.GIU}"
:OPEN15,8,15,"I0":B=EA
-SA:IN$="0":+IN$:IFOP
THEN810
760 OPEN1,8,8,IN$+"P,W"
:GOSUB860:IF A THEN220
770 AH=INT(SA/256):AL=SA-
(AH*256)
:PRINT#1,CHR$(AL):CHR$(AH);
780 FORI=0TOB:PRINT#1,CHR$(PEEK
(BS+I)):IF ST THEN800
790 NEXT:CLOSE1:CLOSE15:GOTO940
800 GOSUB1060:PRINT"{CUR.GIU}
{BLACK}ERRORE DURANTE
IL SAVE:{GRAY1}"
:GOSUB860:GOTO220
810 OPEN1,8,8,IN$+"P,R"
:GOSUB860:IFATHEN220
820 GET#1,A$,B$:AD=ASC(A$
+Z$)+256*ASC(B$+Z$):IF
AD<>SA THENF=1:GOTO850
830 FORI=0TOB:GET#1,A$:POKE
BS+I,ASC(A$+Z$):IF(I<>B)
AND ST THENF=2:AD=I
840 NEXT:IF ST<>64 THENF=3
850 CLOSE1:CLOSE15:ON ABS

```

```

(F>0)+1GOTO960,970
860 INPUT#15,A,A$:IF THEN
CLOSE1:CLOSE15:GOSUB1060
:PRINT"{RVSON}ERRORE:"A$
870 RETURN
880 POKE183,PEEK(FA+2):POKE187,
PEEK(FA+3):POKE188,PEEK
(FA+4):IFOP=0THEN920
890 SYS 63466:IF(PEEK(783)
AND1)THEN GOSUB1060
:PRINT"{CUR.GIU}{RVSON}
FILE NOT FOUND
":GOTO690
900 AD=PEEK(829)+256*PEEK
(830):IF AD<>SATHENF=1
:GOTO970
910 A=PEEK(831)+256*PEEK(832)-
1:F=F-2*(A<EA)-
:AD=A-AD:GOTO930
920 A=SA:B=EA+1:GOSUB1010
:POKE780,3:SYS63338
930 A=BS:B=BS+(EA-SA)+1:
GOSUB1010:ON OP
GOTO950:SYS 63591
940 GOSUB1080:PRINT"{BLUE}
** SAVE COMPLETATO **"
:GOTO220
950 POKE147,0:SYS 63562
:IF ST>0 THEN970
960 GOSUB1080:PRINT"{BLUE}
** LOAD COMPLETATO **"
:GOTO220
970 GOSUB1060:PRINT"{BLACK}
{RVS ON}ERRORE DURANTE
IL LOAD:{CUR.GIU}{GRAY
1}":ON FGOSUB980,990,
1000:GOTO220
980 PRINT"INDIRIZZO DI
PARTENZA ERRATO (";GO
SUB360:PRINT)":RETURN
990 PRINT"LOAD FINITO"A:AD=
SA+AD:GOSUB360:PRINT
D$:RETURN
1000 PRINT"FINITO ALL'INDI
RIZZO FINALE":RETURN
1010 AH=INT(A/256):AL=A-
(AH*256)
:POKE193,AL:POKE194,AH
1020 AH=INT(B/256):AL=B-
(AH*256)
:POKE174,AL:POKE175,AH
:RETURN
1030 IF AD<SA OR AD>EA
THEN1050
1040 IF (AD>511 AND
AD<40960) OR
(AD>49151 ANDAD<53248)
THENGOSUB1080:F=0:RETURN
1050 GOSUB1060:PRINT"{RVSON}
INDIRIZZO NON VALIDO
{CUR.GIU}{BLACK}":F=1
:RETURN
1060 POKESD+5,31:POKESD+6,208:POKE
SD,240:POKESD+1,4:POKESD+4,33
1070 FORS=1TO100:NEXT:GOTO1090
1080 POKESD+5,8:POKESD+6,240
:POKESD,0:POKESD+1,90
:POKESD+4,17
1090 FORS=1 TO100:NEXT
:POKESD+4,0:POKESD,0:
POKESD+1,0:RETURN

```


HAI SCELTO IL MEGLIO



Leader del mercato da oltre dieci anni, il Gruppo Editoriale Jackson ti offre tanta esperienza, informazione e aggiornamento in ciascuna delle sue quattro aree editoriali. Hai scelto il meglio. Ma non fermarti qui. Scopri anche le altre riviste Jackson e, con l'abbonamento, scoprirai anche il piacere di riceverle a casa tua, comodamente, a prezzi eccezionali. E c'è di più: abbonandoti ad una rivista Jackson godrai di uno sconto esclusivo del 10% sulle grandi opere e i libri del Gruppo Editoriale Jackson acquistati direttamente presso l'editore.

HAI SCELTO JACKSON



**GRUPPO EDITORIALE
JACKSON**



PER I TUOI ACQUISTI RIVOLGITI AI PUNTI VENDITA UFFICIALI COMMODORE

LOMBARDIA

MILANO ALPHA COMPUTER Via Tavazzano 14 • **AL RISPARMIO** V.le Monza 204 • **BCS** Via Mantegani 11 • **BRAHA ALBERTO** Via P. Capponi, 5 • **E.D.S.C.** so Porta Ticinese, 4 • **E.S.C.** Via Roggia Scagna, 7 • **FAREF** Via A. Volta, 21 • **FLOPPERIA** SRL V.le Montenero, 31 • **GIGLIONI LAURA** Via D'Ovidio, 8 • **GIGLIONI V.** le Luigi Sturzo, 45 • **LOGITEK** Via Golgi, 60 • **MARCUCCI** Via Fratelli Bronzetti, 37 • **MELCHIONI** Via P. Colletta, 37 • **MESSAGGERIE MUSICALI** Galleria Del Corso, 2 • **NEWEL** Via Mac Mahon, 75 • **RIVOLA** Via Vitruvio, 43 • **S.ANGELO LODIGIANO FERRARI** LUIGI Via Madre Cabrini, 44 • **BARLASSINA F.LLI GALIMBERTI** Via Nazionale dei Giovani, 28/36 • **BOVISIO MASCIAGO R & C** so Milano, 118 • **CINISELLO BALSAMO** GBC V.le Matteotti, 66 • **MILANO** GBC Via Petrella, 6 • **MILANO** GBC Via Cantoni, 7 • **COLOGNO MONZEESE** CASA DELLA MUSICA Via Indipendenza, 21 • **CORBETTA** PENATI Via Verdi, 28/30 • **CORSICO** EPM System V.le Italia, 12 • **DESIO** P. GIORGIO STELLARI Via Milano, 300 • **LEGNANO** CENTROCOMPUTER PANDOLFI Via Corridori, 18 • **LISSONE** COMPUTEAM Via Vecellio, 41 • **LODI** FUTURA Via Solferino, 31 • **LODI M.B.M.C.** so Roma, 112 • **MELEGNANO** L'AMICO DEL COMPUTER SAS V.le Lombardia, 17 • **MONZA** BIT 84 Via Italia, 4 • **NOVATE MILANESE** IL CURSORE Via Cavour, 35 • **OPERA** I.C.O. Via dei Tigli, 14 • **SESTO SAN GIOVANNI** NIWA HARD & SOFT Via Bruno Buozzi, 94 • **VIMODRONE BERGAMO** IL COMPUTER SERVICE SHOP Via Padana Superiore, 197 • **BERGAMO** COMIF Via Autolinee, 10 • **CORDANI** Via dei Caniana, 8 • **D.R.B.** Via Borgo Palazzo, 65 • **NEW SYSTEMS POINT** Via Paglia, 36 • **ALZANO LOMBARDO** BERTULEZZI GIOVANNI Via Fantoni, 48 • **CARVICO** COMPUTER TEAM hi-tec Via Verdi, 1/B • **LOVERE** OTTICO OPTOMETRISTA ROVETTA P.zza Garibaldi, 6 • **SAN PELLEGRINO TERME** A.I.S. INTERNATIONAL Via S. Carlo, 25 • **SARNICO** COMPUTER POINT Via Lantieri, 52 • **BRESCIA** SISTHEMA Via Roma, 45 • **URGNANO BRESCIA E PROVINCIA** A.B. INFORMATICA Strada Statale Cremasca, 66 • **BRESCIA** COMPUTER CENTER Via Cipro, 62 • **INFORMATICA 2000** Via Stazione, 16/B • **MASTER INFORMATICA** Via F.lli Ugolini, 10/8 • **VIGASIO** MARIO Portici Zanardelli, 3 • **BRENO** MISTER BIT Via Mazzini, 70 • **CASTREZZATO** CAVALLI PIETRO Via 10 Giornate, 14/B • **CHIARI** (BS) VIETTI GIUSEPPE Via Milano, 1/B • **DESENZANO DEL GARDA** MEGABYTE P.zza Malvezzi, 1 • **GHEDI DITTA** BARESI RINO & C. Via XX Settembre, 7 • **GRATACASOLO** INFO CAM Via Provinciale, 3 • **COMO** IL COMPUTER Via Indipendenza, 90 • **2M ELETTRONICA** Via Sacco, 3 • **BARZANO** ELTRONGROS VIA L. Da Vinci, 4 • **CASSAGO BRIANZA** EGA Via Mazzini, 42 • **ERBA** DATA FOUND computer shop Via A. Volta, 4 • **LECCO** CIMA ELETTRONICA Via Leonardo Da Vinci, 7 • **FUMAGALLI** Via Cairoli, 48 • **OLGIATE COMASCO** RIGHI ELETTRONICA Via G. Leopardi, 26 • **CREMONA** MONDO COMPUTER Via Giuseppe, 11/B • **PRISMA** Via Buoso da Novara, 8 • **TELCO** P.zza Marconi, 2/A • **CREMA** ELCOM/GBC Via VI Novembre, 56/58 • **EUROELETTRONICA** Via XX Settembre, 92/A • **MANTOVA E PROVINCIA** COMPUTER SAS Galleria, 7 • **32 BIT** (Computer Studio) Via Cesare Battisti, 14 • **ELETTRONICA** DI BASSO V.le Risorgimento, 69 • **PAVIA** POLIWARE SRL C.so Carlo Alberto, 76 • **VIGEVANO** LOGICA INFORMATICA V.le Monte Grappa, 34 • **M. VENTIN** C.so Vittorio Emanuele, 76 • **SONDRIO** CIPOLLA MAURO Via Tremogge, 25 • **SAN PIETRO DI BERBENNO** FOTONOVA V. Valeriana, 1 • **VARESE** DIMECO SISTEMI Via Garibaldi • **IL CENTRO ELETTRONICO** Via Morazzone, 2 • **SUPERGAMES** Via Carrobbio, 13 • **BUSTO ARSIZIO** BIT Via Gavinana, 17 • **MASTER PIX** SNC Via S. Michele, 3 • **CASTELLANZA** CRESPI GIUSEPPE & C. V.le Lombardia, 59 • **GALLARATE** PUNTO UFFICIO Via Raffaello Sanzio, 8 • **GEMONIO** SIDALCO Via Caprera, 10 • **GERENZANO** (VA) GRANDI MAGAZZINI BOSSI Via Cienici, 196 • **SESTO CALENDE** (VA) J.A.C. nuove tecnologie Via Matteotti, 38

PIEMONTE

ALESSANDRIA BIT MICRO Via Mazzini, 102 • **SERVIZI INFORMATICI** Via Alessandro III, 53 • **TORTONA** S.G.E. ELETTRONICA Via Bandelio, 19 • **ASTI** RECORD C.so Alfieri, 166/3 • **ASTICUNEO** ROSSI COMPUTERS C.so Nizza, 42 • **CUNEO** STUDIO SOFTWARE C.so Nizza, 4 • **PUNTO BIT** C.so Langhe, 26/C • **ALBA** SDI Via Vittorio Emanuele, 250 • **BRA** • **FOSSANO** ASCHIERI GIANFRANCO C.so Emanuele Filiberto, 6 • **BOSETTI** Via Roma, 149 • **NOVARA** ELCOM SRL C.so Mazzini, 11 • **PROGRAMMA 3** V.le Buonarroti, 8 • **PUNTO VIDEO** C.so Risorgimento, 39/8 • **ARONA** COMPUTER Via Monte Zeda, 4 • **BORGOMANERO** ALL COMPUTER C.so Garibaldi, 106 • **DOMODOSSOLA** MICROLOGIC Via Giovanni III, 2 • **INTRA** ELLIOTT COMPUTER SHOP Via Don Minzoni, 32 • **TORINO** ABA ELETTRONICA Via C. Fossati, 5/P • **ALEX** COMPUTER E GIOCHI C.so Francia, 333/4 • **VILLASTELLONE** CDM ELETTRONICA Via Marrocchetti, 17 • **spedizione**: C.so Savona, 43 • **TORINO** COMPUTER HOME SNC Via San Donato, 46/D • **COMPUTING NEW** Via Marco Polo, 40/E • **DE BUG** C.so Vittorio Emanuele II, 22 • **DESME** UNIVERSAL Via San Secondo, 95 • **F.D.S.** ALTERIO Via Borgaro, 86/D • **IL COMPUTER** Via Nicola Fabrizi, 140 bis • **INFORMATICA ITALIA** C.so Re Umberto, 129 (128) • **MT INFORMATICA** C.so Giulio Cesare, 58 • **MUSIC** 'S SHOP C.so Potenza, 177 • **NEWBUSINESS** COMPUTER Via Nizza, 45/F • **PLAYGAMES** SHOP Via Carlo Alberto, 39/E • **RADIO TV MIRAFIORI** C.so Unione Sovietica, 381 • **SMT ELETTRONICA** Via Bibiana, 83/B • **TELERITZ** C.so Traiano, 34 • **CHIERI** PAUL E CHICO VIDEO SOUND Via Vittorio Emanuele, 52 • **CIRIE** BIT INFORMATICA Via Vittorio Emanuele, 154 • **COLLENGO** HI-FI CLUB C.so Francia, 92/C • **FAVRIA** MISTER PERSONAL Via Cattaneo, 52 • **IVREA** C.S.S. Stradale Torino N, 73 • **MONCALIERI** BAS C.so Roma, 47 • **PINEROLO** CERUTTI MAURO C.so Torino, 234 • **RIVAROLO** EUREX C.so Indipendenza, 5 • **RIVOLI** C.SE DIAM INFORMATICA C.so Francia, 1 46/bis • **FULL INFORMATICA** Via Vittorio Veneto, 25 • **SAN MAURO TORINESE** PANORAMA TORINO SPA Strada Settimo, 371 • **SETTIMO TORINESE** GAMMA COMPUTER Via Cavour, 3 A/B • **VERCELLI E PROVINCIA** DITTA ELETTRONICA C.so Bormida, 27 • **ELETTRONICA** Strada Torino, 15 • **BIELLA** C.S.I. TEOREMA Via Losana, 9 • **SIGEST** SRL Via Bertodano, 8 • **BORGOSIESA** REMONDINO FRANCO Via Roma, 5 • **COSSATO** FOTOSTUDIO TREVISAN Via XXV Aprile, 24/B • **TRINO** STUDIO FOTOGRAFICO IMARISIO P.zza Martiri Libertà, 7

VENETO

BELLUNO UP TO DATE di VIEL RENZO Via Vittorio Veneto, 43 • **PADOVA E PROVINCIA** GUERRA COMPUTERS FELTRE Via Mazzini, 10/C • **PADOVA** BIT SHOP Via Cairoli, 11 • **COMPUMANIA** Riviera Tisa da Camposampiero, 37 • **COMPUTER POINT** Via Roma, 63 • **D.P.R.** Vicolo Lombardo, 4 • **GIANFRANCO** MARCATO Via Madonna della salute, 51/53 • **spedizione**: Via Bergamini, 4 • **SARTO** COMPUTER Via Armistizio, 79 • **ZELLA ADELIO** P.zza De Gasperi, 31/A • **CITTADELLA** RIVIGO COMPUTER SERVICE Borgo Treviso, 150 • **RIVIGO** TREVISIO CLINICA DEL RASOIO E DEL COMPUTER Via Fiume, 31/33 • **TREVISO** BIT 2000 Via Brandolini D'Adda, 14 • **GUERRA** EGIDIO & C. V.le Cairoli, 95 • **CONEGLIANO** DE MARIN COMPUTERS Via XX Settembre, 74 • **MONTEBELLUNA** SIDESTREET Via Salvo D'Acquisto, 8 • **PREGANZIOL** FALCONE ELETTRAUDIOVIDEO Via Terraggio, 116 • **VENEZIA E PROVINCIA** TELERADIO Fuga San Marco • **spedizione**: IVANO BERTOLA Via Rossi Rubano • **MESTRE-VENEZIA** TREKILOWATT Via Torre Belfredo, 47 • **CHIOGGIA** T.C.H. Riva Vena, 889 • **SAN DONA' DI PIAVE** (VE) GUERRA COMPUTER Via Vizzotto, 29 • **REBEL** Via F. Crispi, 10 • **SOTTOMARINA** TELFRET di Ferretto Flavio Via Chiesa • **SPINEA** RADIOCESTARO Via Roma, 89 • **VERONA** CASA DELLA RADIO Via Cairoli 10 • **TELESAT** Via Vasco De Gama, 8 • **spedizione**: Ivano Bertola Via Rossi • **LEGNANO** RUBANO FERRARIN Via dei Massari, 10 • **spedizione**: Ivano Bertola Via Rossi • **VICENZA** ELETTRONICA BISELLO V.le Trieste, 427/429 • **SCALCHI** MARKET Via Ca Balbi, 139 • **CECCATO** GUERRA COMPUTER Via Dell'Industria Alte • **CAVAZZALE** SCHIAVOTTO Via Zanella, 21 • **COMPUTER** B.COSTO Via del Costo, 34

FRIULI VENEZIA GIULIA

GORIZIA THIENE ELETTRONICA Via Roma, 67 • **GORIZIA PORDENONE E C.O.** ELETTRONICA COMMERCIALE Via F.lli Cossar, 23 • **PORDENONE** RIGO V.le Cossetti, 5 • **BRUNO** DA PIEVE & C. Via Colombara, 17 • **VILLANOVA DI PRATA** TRIESTE PORCIA MDT P.zza De Repubblica, 5 • **TRIESTE** AVANZO GIACOMO P.zza Caviana, 7 • **COMPUTER SHOP** Via P. Reti • **TRIESTE** COMPUTIGI Via XX. Settembre, 51 • **TRIESTE** UDINE CT Via Pascoli, 4 • **UDINE** MOFERT V.le Europa Unità, 4 • **spedizione**: Mifert, 2 Via Leopardi 2 • **R.T.** SISTEM UDINE Via L. Da Vinci, 99 • **MARTIGNACCO** ANDRENO MATTIUSI & C. Via Liciniana, 58

TRENTINO ALTO ADIGE

BOLZANO COMPUTER POINT Via Roma, 82A • **BOLZANO** MATTEUCCI PRESTIGE Via Museo, 54 • **spedizione**: ELETTRON MATTEUCCI Via Parma • **BRUNICO** RADIO MAIR-ELECTRO Via Centrale 70 • **MERANO** ELECTRO RADIO HENDRICH Via Delle Corse, 106 • **SILANDRO** (BZ) TRENTO ELECTRO TAPPEINER P.zza Principale, 90 • **TRENTO** CRONST SAS Via G. Galilei, 25 • **ELETTRONICA** V.le Verona, 9

LIGURIA

GENOVA ABM COMPUTER P.zza De Ferrari, 24 rosso • **GENOVA SESTRI PONENTE** CENTRO ELETTRONICO Via Chiaravagna, 10R • **GENOVA** COMMERCIALE SOTTORIPA Via Sottoripa, 115/117 • **SAMPIERDARENA** GENOVA COMPUTER Via D.G. Storace 4/rosso • **GENOVA** FOTOMONDIAL Via Del Campo 3-5-9-11-13 r • **LA NASCENTE** Via San Luca, 4/1 • **GENOVA IMPERIA** RAPPR-EL Via Borghetti, 23/R • **IMPERIA** CASTELLINO Via Belgrano, 44 • **spedizione**: SASA COMPUTER • **SANREMO** (Terzo) CENTROHI-FI VIDEO Via della Repubblica, 38 • **VENTIMIGLIA** CASTELLINO Via Genova, 48 • **spedizione**: SASA COMPUTER (Terzo) • **LA SPEZIA E PROVINCIA** I.L. ELETTRONICA Via Vittorio Veneto, 123 • **FORNOLA DI VEZZANO** SAVONA I.L. ELETTRONICA Via Aurelia, 299 • **SAVONA** ATHENA INFORMATICA Via Carissimo e Crotti, 16/R • **CASTELLINO** C.so Tardy e Benec, 101 • **spedizione**: SASA COMPUTER (Terzo)

EMILIA ROMAGNA

PIACENZA COMPUTER LINE Via G. Carducci, 4 • **DELTA** COMPUTER Via Martiri della Resistenza, 15/G • **SOVER** Via VI Novembre, 60

TOSCANA

AREZZO DELTA SYSTEM Via Piave, 13 • **FIRENZE** ATEMA Via Benedetto Marcello, 1a-1b • **ELETTRONICA** CENTOSTELLE Via Cento Stelle, 5/a-b • **HELP** COMPUTER Via degli Artisti, 15-A • **PUNTO** SOFT Via Vagnetti, 17 • **TELEINFORMATICA** TOSCANA Via Bronzino, 36 • **EMPOLI** WAR GAMES SDF Via Raffaello Sanzio, 126/A • **FIGLINE** VALDARDO (FI) NEW E.V.M. COMPUTER Via Degli Innocenti, 2 • **PONTASSIEVE** CENTRO INFORMATICA Via Znojmo, 41 • **PRATO** (FI) COSCI F.lli Via Roma, 26 • **GROSSETO** COMPUTER SERVICE Via Dell'Unione, 7 • **LIVORNO** ETA BETA VIA San Francesco, 30 • **FUTURA** 2 Via Cambini, 19 • **PIOMBINO** ELETTRONICA ALESSI Via Cimarosa, 1 • **LIDO DI CAMAIORE** IL COMPUTER V.le Colombo, 21 6 • **S. ROMANO GARFAGNANA** (LU) SANTI VITTORIO Via Roma, 23 • **MASSA** EURO COMPUTER P.zza B. Bertagnini, 4 • **FIRMIWARE** Galleria Pregliasco, 17 • **CARRARA** (MS) RADIO LUCONI Via Roma, 24/B • **PISA** C.H.S. SNC Via Carlo Cattaneo, 90/92 • **ELECTRONIC** SERVICE Via Della Vecchia Trancia, 10 • **IT - LAB** Via Marche, 8A-8B • **ditta** TONY HI-FI Via G. Carducci, ang. Canto Del Nicchio, 2 • **PISTOIA E PROVINCIA** ELECTRONIC SHOP Via Della Madonna, 49 • **OFFICE DATA** SERVICE Galleria Nazionale, 22 • **MONTECATINI** TERME ZANNI & C. C.so Roma, 45 • **TEL.** 0572-79649 • **spedizione**: Via Forini, 10 • **SIENA** VIDEO MOVIE Via Garibaldi, 17 • **CHIANCIANO TERME** ELECTRIC SHOP Via A. Casini, 51 • **MONTEPULCIANO** ELETTRONICA Via di Graccia-no nel Corso, 111

UMBRIA

PERUGIA LA MIGLIORATI Via S. Ercolano, 310 • **BASTIA UMBRA** COMPUTER STUDIO'S VIA IV Novembre, 18/A • **CITTA' DI CASTELLO** WARE Via Dei Casceri, 31 • **MATERA** GIOVANNI GAUDIANO ELECTRONICS Via Roma

PUGLIA

BARI ARTEL Via Guido D'Orso, 9 • **COMPUTER'SARTS** V.leb Meucci, 12/B • **PAULICELLI** SABINO & FIGLI Via Fanelli, 231/C • **BARLETTA** F. FAGGELLA C.so Garibaldi, 15 • **GIANNI FAGGELLA** P.zza D'Aragona, 62A • **CASTELLANA** LONUZZO GIUSEPPE Via Nizza, 21 • **BRINDISI** MARANGI E MICCOLI Via Prov. San Vito, 165 • **FRANCAVILLA FONTANA** MILONE GIOVANNI Via San Francesco D'Assisi, 219 • **FOGGIA** BOTTICELLI GUIDO Via San Pollice, 2 • **E.C.I.** COMPUTER Via Isonzo, 28 • **LA TORRE** SAS V.le Michelangelo, 185 • **SAN SEVERO** IL DISCOBOLO Via T. Solis, 15 • **LECCE** BIT Via 95° Reggimento Fanteria, 87/89 • **TRICASECEDOK** INFORMATICA Via Roma, 31 • **TARANTO** FERNANDO DE SANTIS Via Muro Maglie • **ELETTROJOLLYCENTRO** Via De Cesare, 13 • **TEA** Via Regina Elena, 101

CAMPANIA

ATRIPALDA FLIP FLOP Via Appia, 68 • **BENEVENTO** E.CO. INFORMATICA Via Pepicelli, 21/25 • **CASERTA** O.P.C. Via G.M. Bosco, 24 • **MADDALONI** M.P. COMPUTER Via Napoli, 30 • **NAPOLI** BABY TOYS Via Cisterna Dell'Olio, 5/bis • **CASA MUSICALE** RUGGIERO Int. Stazione Napoli • **C.LE F.S.** P.zza Garibaldi, 74 • **CENTRO ELETTRONICO** CAMPANO Via Epomeo, 121 • **CIAN** Galleria Vanvitelli, 32 • **CINE NAPOLI** SNC Via Santa Lucia, 93/95 • **DARVIN** Calata San Marco, 26/25 • **ELETTRONICA** RO.DA.LO. Via Epomeo, 216/B • **GIANCAR** 2 P.zza Garibaldi, 37 • **GRUPPO** BUSCH Galleria Umberto, 55 • **ODORINO** L.go Lala, 22/A-B • **R 2 SRL** Via F. Cilea, 285 • **TOP VIDEO** - TOP COMPUTER Via S. Anna Dei Lombardi, 12 • **SPY** Via San Giacomo Dei Capri, 36/B • **VIDEOFOTOMARKET** Via S. Brigida, 19 • **S. ANASTASIA** (NA) SPADARO Via Romani, 93 • **CASORIA** ELECTRONIC DAY Via Delle Puglie, 17 • **TUFANO** S.S. Sannitica, 87 • **CASTELAMARE** DI STABIA SOF SUD V.le Europa, 59 • **FRATTAMAGGIORE** ELETTRONICA 2000 C.so Durante, 40 • **MUGNANO** GATEWAY Via Napoli, 68 • **PORTICI** NUOVA INFORMATICA SHOP Via Libertà, 185/191 • **POZZUOLI** BASIC COMPUTER C.so Garibaldi, 34 • **STRIANO** FALCO ELETTRONICA Via Sarno, 100 • **TORRE ANNUNZIATA** TECNOTRE Via P. Fusco, 1/F • **TORRE DEL GRECO** (NA) TECNO Via Vittorio Veneto, 28 • **SALERNO** COMPUTER MARKET C.so Vittorio Emanuele, 23 • **BATTIPAGLIA** (SA) KING COMPUTER Via Olevano, 56 • **EBOLI** (SA) DIMER POINT Via C. Rosselli, 20

CALABRIA

CATANZARO C. & G. COMPUTER Via F. Acri, 28 • **PAONE** SAVERIO & FIGLI Via F. Acri, 93/99 • **CROTONE** COMPUTER HOUSE Via Bologna (L.go Ospedale) • **VIBO VALENTIA** OTTICA FOTO NELLO RUELLO C.so Vittorio Emanuele, 177 • **COSENZA** SIRANGELO COMPUTER Via Parisio, 25 • **HI-FI** ALFANO GIUSEPPE Via Baidacchini, 109 • **CASTROVILLARI** AMANTIA ELIGIO ANNICCHIARICO & C. SAS Via Roma, 21 • **CORIGLIANO** SCALO ALFA COMPUTER Via Nazionale, 341/A • **LAMEZIA TERME** ING. FUSTO SALVATORE C.so Nicotera, 99 • **REGGIO CALABRIA** CONTROL SYSTEM Via San Francesco Da Paola, 49/D/E • **SYSTEM HOUSE** Via Fiume ang. Palestino, 1 • **LOCRI** (RC) COMPUTER SHOP V.le Matteotti, 50-52 • **TAURIANOVA** PICIEFFE SNC C.so F. Sofia Alessio

ESIGI SEMPRE LA GARANZIA DELLA COMMODORE ITALIANA

SUPERCOMMODORE N.26

SERVIZIO LETTORI **Compilare e spedire in busta chiusa a: GRUPPO EDITORIALE JACKSON**
Area Consumer - Via Rosellini, 12 - 20124 Milano

A) Come giudichi questo numero di Supercommodore?

- ☐ Ottimo
☐ Molto Buono
☐ Buono
☐ Discreto
☐ Sufficiente
☐ Insufficiente

B) Quale (il) articolo (i) o rubrica hai apprezzato di più?

Quale meno?

C) Cosa ti piacerebbe leggere nei prossimi numeri di Supercommodore?

E) Quante persone leggono la tua copia di Supercommodore?

F) Che computer possiedi?

Quale (il) computer intendi acquistare in futuro?

G) Leggi altre riviste Jackson?

☐ SI ☐ NO

Quali?

H) Leggi altre riviste dedicate al tuo computer?

☐ SI ☐ NO

Quali?

I) Oltre alle riviste dedicate al computer quali sono le tue letture preferite?

L) Quali sono i tuoi hobbies e maggiori interessi?

- ☐ Sport ☐ Fotografia
☐ Musica ☐ Automobile
☐ Videoregistrazione ☐ Moto
☐ Hi-Fi ☐ Viaggi

Altro

Nome

Cognome

Indirizzo

Età

Professione

Città

Prov.

C.a.p.

Tel.

Fondamentali per lo studio, il lavoro e l'aggiornamento i dizionari enciclopedici di:

**Matematica
Fisica • Chimica
Informatica • Meccanica
Astronomia • Biologia • Geologia
Ragioneria Generale
Ragioneria Applicata • Elettronica**

IN EDICOLA
OGNI MESE QUATTRO ARGOMENTI
A LIRE 14.000 CIASCUNO



Conoscenza e
informazione, chiarezza e
rigore scientifico in
15.000 termini e oltre
650 illustrazioni, tabelle e schemi.

Fondamentali per il nostro tempo.



GRUPPO EDITORIALE
JACKSON

NOVITA' ASSOLUTA IN EDICOLA

Guida VIDEO GIOCHI

L. 3.500

LA GRANDE GUIDA A TUTTI I GIOCHI ELETTRONICI E NON

Nuovissima, ricca e tutta a colori. GUIDA VIDEOGIOCHI ti aspetta in edicola con oltre 60 giochi recensiti, i commenti, le curiosità, i trucchi e le novità da tutto il mondo.

E, in più, partecipi al grande concorso riservato ai fedeli lettori di GUIDA VIDEOGIOCHI.

FANTASTICO CONCORSO
GUIDA VIDEOGIOCHI

I premi
in palio sono
favolosi: due esclusive
Control Deck NINTENDO
e tanti game originali.

Nintendo



GRUPPO EDITORIALE
JACKSON

